Facultad
de Matemática,
Astronomía, Física
y Computación

Estratificación temporal de Aedes Aegypti basada en herramientas geoespaciales y aprendizaje automático

Juan Manuel Scavuzzo

Universidad Nacional de Córdoba
Facultad de Matemática, Astronomía, Física y Computación
Córdoba, Argentina
2018

Estratificación temporal de Aedes Aegypti basada en herramientas geoespaciales y aprendizaje automático

Juan Manuel Scavuzzo

Trabajo final de grado presentado como requisito parcial para optar al título de:
Licenciado en Ciencias de la Computación

Directores:
Dr. Jorge Sánchez (Ingeniero en Electrónica y Doctor en Ciencias de la Ingeniería, Visión por computadoras y reconocimiento de patrones)
Mgter. Gonzalo Sebastián Peralta (Licenciado en Cs de la Computación y Magíster en Aplicaciones Espaciales)

Universidad Nacional de Córdoba
Facultad de Matemática, Astronomía, Física y Computación
Córdoba, Argentina
2018

Agradecimientos

Soy de las personas que piensan que los logros son colectivos. Por más que los títulos estén a nombre de una sola persona, no sería posible realizar logros plenamente individuales, y si lo fuera, éstos no serían lo mismo... tendrían vacíos, les faltaría algo. Los logros colectivos nos permiten entender y aprender de manera integral. Nos enseñan a que siempre tenemos algo que aprender del otro y tenemos algo que convidar para el otro... Para mí, un buen profesional es aquel que comprende que tiene algo que aportar a la sociedad (y muchas cosas que aprender de ella), ya sea desde lo técnico o desde la capacidad adquirida para razonar sobre cuestiones cotidianas.

Es por todas las razones que menciono, que agradezco plenamente a todas las personas que pasaron, durante todos estos años, por mi vida. Nombrar uno por uno los nombres, quizás llevaría demasiado y por eso me toca hacer énfasis y mencionar a aquellas que estuvion más cerca durante todo este proceso.

Quiero agradecer a mis viejos, Trinidad y Marcelo, quienes con todo su esfuerzo me brindaron la posibilidad de dedicarme a estudiar y enfocar mis esfuerzos en aquello que yo elegí como proyecto de vida, siempre apoyándome en cualquier decisión tomada.

A Lula, mi compañera de vida, mi cómplice y mejor amiga. Quién me apoyó, aguantó y disfrutó los últimos años de este gran camino recorrido. Sin dudas, fue de las personas que me llenó de fuerzas en los últimos tramos, cuando la energía escasea.

A mis hermanos, Marco y Matías, que desde siempre estuvieron ahí acompañando y apoyando por más que también estuvieran haciendo su camino al andar, con todas las dificultades que eso implica.

A Gonzalo, quien desde su lugar cercano a la familia, es uno de los responsables de que haya elegido esta carrera. Quien, a pesar del cariño, fue mi docente y director de tesis pudiendo exigirme para verme crecer.

A mis amigos-compañeros hechos en las clases, Trucco, Agus, Fran, Marcos y Limón con quienes pasamos muchas andanzas y horas de estudio. Sin los cuales, hubiera sido de gran dificultad afrontar todas las dificultades de la carrera. Son gente que me llevo para las próximas etapas que toque vivir.

Resumen

Palabras clave: Computer Science, Machine Learning, Python, Landscape Epidemiology, Remote Sensing, Dengue, Zika, Chikungunya, Public Health.

El Dengue, Zika y Chikungunya son enfermedades virales cuya vacuna para prevención aún no existe y que, en los últimos años, han tenido un incremento e impacto en la población de la región argentina y latinoamericana. Razón por la cual, son una gran preocupación para los organismos gubernamentales de salud.

En los últimos años se han generado sistemas para la estimación de riesgo de transmisión de enfermedades virales basados en información de sensores remotos, estableciendo relaciones entre las condiciones ambientales de las distintas zonas y la abundancia del vector en las mismas. A dicha área de estudio se la denomina Epidemiología Panorámica.

En el presente trabajo, por un lado, se utilizan técnicas de ingeniería del software para extraer los requerimientos y aplicar una metodología de desarrollo acorde a las necesidades, para implementar un *framework* para la generación de modelos de aprendizaje automático (ML) con el objetivo de estimar la abundancia de vectores de Dengue, Zika y Chikungunya. A su vez, se entrenan y evalúan modelos no lineales para modelar las poblaciones del mosquito. Éstos poseen mayor capacidad de generalización, en comparación con los modelos que actualmente se utilizan para tal fin.

Se presenta, además, un enfoque que resuelve el problema de que un modelo entrenado con información de una sola ciudad no es capaz, en principio, de estimar correctamente la abudancia en otras zonas del país. En este trabajo se propone resolver la cuestión a traves de un concepto novedoso en el campo de la epidemiología, que establece relaciones de “cercanía“ entre regiones teniendo en cuenta sus características ambientales: la Distancia Ambiental Normalizada.

Abstract

Dengue, Zika and Chikungunya are viral diseases whose vaccine for prevention does not exist yet and which, in recent years, have had an increase and impact on the population of the Argentine and Latin American region. Which is why they are a major concern for government health agencies.

In recent years, systems have been generated to estimate the risk of transmission of viral diseases based on information from remote sensors, establishing relationships between the environmental conditions of the different zones and the abundance of the vector in them. This area of study is called Landscape Epidemiology.

In the present work, on the one hand, software engineering techniques are used to extract the requirements and apply a development methodology according to the needs to implement a *framework* for the generation of machine learning models (ML) with the objective of estimating the abundance of Dengue, Zika and Chikungunya vectors. In turn, non-linear models are trained and evaluated to model mosquito populations. These have greater generalization capacity, in comparison with the models that are currently used for this purpose.

It also presents an approach that solves the problem that a model trained with information from a single city is not able, in principle, to correctly estimate the abudance in other areas of the country.

In this work we propose to solve the issue through a novel concept in the field of epidemiology, which establishes relations of " closeness" between regions taking into account their environmental characteristics: the Normalized Environmental Distance.

chapter*

Contenido

1. Motivación y objetivos

1.1. Motivación

El mosquito es uno de los vectores de enfermedades humanas más importantes en el mundo. En particular, el *Aedes aegypti* es el principal vector de Dengue, Chikungunya, Zika y Fiebre Amarilla urbana [50]. Según datos de la Organización Mundial de la Salud (OMS), alrededor de 80 millones de personas se infectan de Dengue anualmente, cerca de 550 mil enfermos requieren hospitalización y unos 20 mil mueren. Además, calculan que más de 2.500 millones de personas corren riesgo de contraer la enfermedad y más de 100 países tienen transmisión endémica [55]. Cabe aclarar que en el caso de la Fiebre Amarilla existe una vacuna de virus atenuado que se considera eficaz para la prevención, segura y se la utiliza hace más de 60 años en la inmunización activa de niños y adultos. Para el Dengue, Chikungunya y Zika, no existe tal herramienta de previsión.

Médicos del Mundo[1], en su nota **"Médicos del Mundo alerta sobre riesgos de fiebre amarilla en Brasil y escenarios de Dengue-Zika en Argentina"** explican que entre 1985 y 2012, si tenemos en cuenta las cuatro enfermedades mencionadas en el párrafo anterior, en las Américas el 95 % de los casos se concentraron en 4 países: Perú (54 % de los casos), Bolivia (18 %), Brasil (16 %) y Colombia (7 %). También dicen que, aún así, Argentina, Ecuador, Panamá y Venezuela también tienen condiciones muy favorables para la transmisión.

En el caso de Argentina, para el año 2016, podemos observar la taza de Dengue a nivel provincia [78] en la Figura **1-1**[2], a su vez desde la emergencia del virus del Zika en nuestro país en el mismo año (Tucumán), y hasta el 2017 se registraron además un total de 7 casos confirmados de síndrome congénito asociados a virus del Zika en mujeres embarazadas (microcefalia en recién nacidos). Durante el 2017, en base a las notificaciones al **Sistema Nacional de Vigilancia de Salud** del Ministerio de Salud de la Nación recibidas hasta el 30 de diciembre, se registraron, en el primer semestre del año, brotes de Dengue serotipo DEN-1 con 646 casos confirmados en cinco provincias (Buenos Aires, Chaco, Corrientes, Formosa y Santa Fe) y 253 casos de enfermedad por virus del Zika en tres provincias (Chaco, Formosa y Salta). Incluso ya para las primeras semanas del 2018, hubo casos confirmados de Dengue en Chaco.

[1] http://www.mdm.org.ar

[2] Figura brindada por Rotela y colaboradores de su trabajo [78].

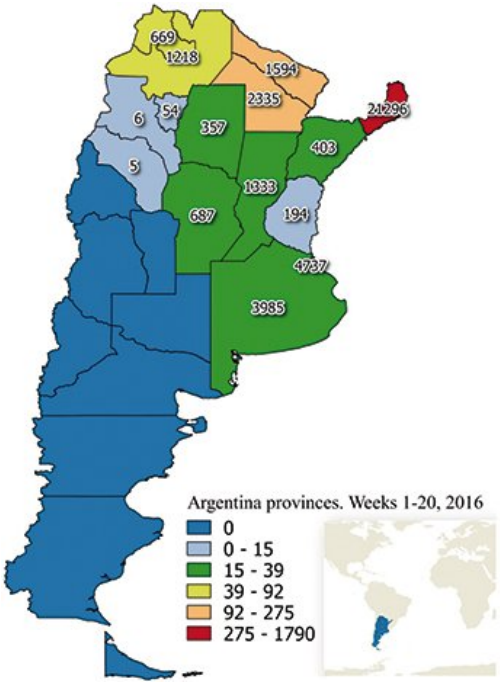


Figura 1-1.: La taza de Dengue en Argentina a nivel provincia en 2016. Dicha taza es expresada cada 100.000 habitantes; se tienen en cuenta todos los casos confirmados y probables hasta la semana 20.

El Dr. Gonzalo Basile [3] se refiere al incremento del riesgo de crecimiento en la cantidad de casos positivos en nuestro país, teniendo en cuenta el contexto epidemiológico en la región:

> Aunque las últimas epidemias del 2009 y 2016 de Dengue en Argentina fueron del serotipo DEN1, la circulación viral de los otros serotipos en la región de Cono Sur (Brasil, Paraguay y Bolivia) tanto DEN4, DEN2 y DEN3, hace que los periodos epidémicos de DEN se puedan modificar. Por otro lado, el escenario de Zika Virus es una realidad por su circulación en América Latina y Caribe con cuadros clínicos inéspecíficos pero con eventos asociados como el Síndrome de Guillaen Barré y microcefalia que implican problemas epidemiológicos poblacionales de incidencia como lo demostraron en Brasil, Colombia, Venezuela, República Dominicana, entre otros 47 países de la región donde se confirmaron casos de transmisión activa vectorial de Zika. Si sumamos ahora el brote epidémico de Fiebre Amarilla en Brasil con la posibilidad de reintroducir casos en el Cono Sur ya que las tasas de inmunizaciones para fiebre amarrilla existen brechas en varias ciudades de nuestro país. **24/01/2018**

[3]Presidente Honor y Director General para América Latina y Caribe de Médicos del Mundo, e investigador de institutos de investigación en salud pública del Caribe y coordinación regional del Programa de Salud Internacional de CLACSO y de FLACSO República Dominicana.

Sumado a lo comentado, por su parte, el *Aedes aegypti* se caracteriza por su presencia en el medio urbano, su preferencia de cría en contenedores artificiales [81] y la resistencia de sus huevos a la desecación. En nuestro país, además, los vertiginosos cambios demográficos, han dado por resultado una gran ampliación desorganizada de las zonas urbanas. Ésto, junto con el aumento del uso de recipientes no biodegradables y un método deficiente de recolección de residuos sólidos, incrementan el número de depósitos que acumulan agua, que actúan como potenciales criaderos del mosquito, lo cual aumenta el riesgo de ocurrencia de casos de las enfermedades mencionadas. Dado que la cantidad de vectores, el virus circulante y la susceptibilidad humana dependen directa o indirectamente de variables climáticas y ambientales tales como la temperatura, la lluvia, la vegetación, entre otras, el cambio climático es, también, un factor de riesgo para el desarrollo de las enfermedades en cuestión. Por otra parte, se le suma la capacidad adaptativa del *Aedes aegypti* y la aparición de resistencia del mismo debido al uso intensivo de insecticidas.

Por otro lado, como se menciona anteriormente, dado que no hay vacunas para la mayoría de estos virus, y existe la posibilidad de introducción de otros [93], el control de vectores es la principal herramienta para mitigar la propagación de enfermedades.

Es claro que el escenario epidémico planteado es una realidad en Argentina que hay que atacar. Ésto deriva en la necesidad de enfocar esfuerzos en el desarrollo de estrategias contundentes dirigidas a evitar, limitar y controlar las poblaciones de *Aedes aegypti*, lo que implica repensar y diseñar nuevos sistemas de alerta temprana, vigilancia epidemiológica y respuesta rápida desde lo local, integrando un espacio interinstitutional e intersectorial de coordinación, planificación e intervención pública. Ésto debe llevarse a cabo entre el Estado Nacional, municipios, universidades y centros de estudio, organizaciones civiles, entre otros actores sociales de gran importancia. En ese contexto, la introducción de herramientas científico/tecnológicas orientadas a contribuir en esos aspectos resulta fundamental.

El uso de información satelital se ha estado utilizando desde hace algunos años, como uno de los métodos para atacar el problema mencionado. Ésta técnica permite modelar la evolución temporal y geográfica de las poblaciones del vector utilizando variables ambientales obtenidas de los sensores remotos. Aunque hasta ahora, estos trabajos utilizaban fuertes suposiciones al utilizar modelos lineales para relacionar las distintas variables, y por más que los resultados obtenidos hasta el momento han sido favorables, no existen resultados que prueben el hecho de que las relaciones que se establecen entre el desarrollo del mosquito y las variables ambientales del entorno son de tipo lineal. Esto nos permite generar la hipótesis de que modelos no-lineales son capaces de modelar relaciones que se adapten mejor a la realidad. Una de las formas de probarla es utilizar modelos no-lineales de aprendizaje automático para llevar a cabo esa tarea.

Desarrollar un modelo de aprendizaje automático puede resultar extremadamente complejo y costoso en términos computacionales y de experiencia de quien lo lleve a cabo. Uno de los objetivos de este trabajo es mostrar la accesibilidad, en términos de simpleza y costos, de algunas de estas herramientas, sin dejar de lado el desempeño en la tarea concreta. A su

vez, también existe el importante problema de la escasez de datos de campo para utilizarlos en la construcción de los modelos. Hasta ahora, era un gran limitante ya que no se tienen datos vitales para el desarrollo de este tipo de herramientas. En este trabajo, además, se propone una técnica para atenuar dicho problema estableciendo una relación entre los distintos puntos geográficos, en función de sus características ambientales.

1.2. Objetivos

Los objetivos del trabajo apuntan a aportar conocimientos y herramientas a los profesionales dedicados al desarrollo de sistemas de prevención y mitigación del Dengue, Zika, Chikungunya y enfermedades vectoriales en general. Para ello, se establecen los siguientes puntos a desarrollar:

- Implementación de una herramienta para la generación de nuevos modelos predictivos a partir de variables ambientales. Dadas las características interdisciplinarias de la problemática en la epidemiología, dicha herramienta debe ser utilizable por profesionales que no sean expertos en informática o en aprendizaje automático.
- Validar la hipótesis de que modelos no-lineales son mejores para predecir y ajustar la oviposición de mosquitos que los modelos lineales utilizados hasta el momento.
- Proponer una solución a la problemática de la escasez de puntos con datos de campo para entrenar los modelos, dada la naturaleza regional de un sistema de riesgo, que aporte a valor los existentes.

2. Marco teórico

2.1. Epidemiología panorámica

La *Teledetección* se define como el proceso de adquirir información acerca de un objeto, área o fenómeno desde la distancia. Un sensor remoto es un instrumento capaz de realizar percepción remota, por lo que en esta amplia definición caben desde los ojos hasta los radiotelescopios.

Existen dos grandes tipos de sensores remotos (SR): activos y pasivos. Los activos son aquellos que obtienen la información generando su propia energía mientras que los pasivos dependen de una fuente externa, que en la Tierra principalmente proviene del Sol. Hasta el día de hoy, los más usados son los sensores pasivos dado que permiten medir la magnitud de la radiación electromagnética reflejada e irradiada desde la superficie de la Tierra y de la atmósfera y, de esta manera, derivar información sobre las condiciones de la superficie [76].

Los SR más utilizados y con mayor cantidad de aplicaciones son los que se encuentran a bordo de satélites que orbitan sobre la Tierra, bien sea en órbitas geoestacionarias[1], u órbitas polares, aquellas que pasan repetidamente por diferentes áreas de la Tierra mientras están orbitando alrededor del planeta a altitudes menores.

Las tecnologías relacionadas al ámbito aeroespacial dieron lugar a programas que integran estas tecnologías con, por ejemplo, la agricultura, salud pública, geología y las ciencias forestales. A su vez, la información obtenida por dichos SR se puede aplicar a estudios entomológicos[2], debido a que ellos proveen gran cantidad y diversidad de información sobre la cobertura de la Tierra: características de la vegetación, cuerpos de agua, temperaturas, entre otras. Ésta, también es información sobre el hábitat de insectos y artrópodos vectores [8, 22], y, por lo tanto, de acuerdo a la teoría de Pavlovsky [66] en la que expone la correlación entre el hábitat y enfermedades transmitidas por vectores, los datos de los SR se pueden utilizar como fuente de información sobre la distribución espacio-temporal de dichas afecciones.

Con la acumulación de datos registrados por sensores remotos desde los años 70 existen series temporales que permiten realizar varios tipos de análisis con relevancia para la transmisión de la enfermedad de Dengue y otras ETV[3]. Entre ellas, series temporales de imágenes de mediana resolución espacial permiten analizar en perspectiva histórica los cam-

[1] Están en altitudes entre 23000 y 40000 km, sobre la franja ecuatorial y viajan a la misma velocidad de rotación de la Tierra por lo que siempre están fijos sobre un punto determinado de la superficie terrestre.
[2] De insectos.
[3] Enfermedades de Transmisión Vectorial.

bios de uso y cobertura del terreno, proceso que habitualmente tiene vinculación con cambios en la epidemiología de la enfermedad [27]. A su vez, el deterioro de las condiciones de salud en el mundo, el avance significativo en el procesamiento de computadoras, la mejora en la adquisición de datos, la reducción de los costos de hardware y software y el desarrollo de tecnología GIS[4](por sus siglas en inglés) han llevado al lanzamiento de programas que apuntan a integrar SR / GIS en aplicaciones de salud [17, 23, 30, 34].

El uso de técnicas de Teledetección para mapear la distribución de vectores y el riesgo de enfermedades ha tenido una gran evolución durante las últimas dos décadas. La complejidad de las técnicas va desde el uso de simples correlaciones entre las firmas espectrales de diferentes coberturas, usos del suelo y abundancia de especies hasta técnicas complejas que integran variables ambientales obtenidas de satélites con la biología de los vectores. Estas técnicas se usan para desarrollar modelos predictivos de riesgo, los cuales principalmente se realizan a través de técnicas estadísticas de regresión logística y análisis discriminante, que dilucidan las asociaciones entre datos ambientales multivariados y los patrones de presencia o ausencia de vectores para así mapear los vectores o las enfermedades. Estos métodos son capaces de predecir la probabilidad "*a posteriori*" de la presencia de la variable dependiente (vector o enfermedad), a partir de un grupo de variables independientes (datos de clima y cobertura de la tierra) y de esta forma pueden ser usados para hacer mapas de riesgo a partir de bases de datos.

Las condiciones ambientales que determinan la conectividad[5] de los paisajes para la dispersión pueden variar en las distintas regiones y dependen de cómo el patógeno se dispersa biológicamente (ej. dado un patógeno portado por vectores, el movimiento del insecto) o abióticamente (e.j: flujos de viento o agua). Por ejemplo, rios y corrientes pueden actuar como corredores de dispersión que fomentan la propagación de la infección a través de paisajes heterogéneos para patógenos de plantas transmitidos por el agua. En otros sistemas, como las enfermedades zoonóticas de mamíferos terrestres, estos mismos cuerpos de agua pueden funcionar como barreras impidiendo el movimiento del huésped o del vector. Estas condiciones se ven reflejadas en la Figura **2-1** de [52]. Notemos que en el caso de **a)**, la conectividad entre los sitios azules es mayor que la que se da entre éstos y los amarillos, y también entre ellos y los rojos, siendo que la distancia euclídea entre los azules es mayor. Ésto ocurre porque el sitio rojo está del otro lado de la cordillera, la cual funciona como una barrera geográfica para la inoculación[6], el huésped y/o la dispersión del vector. En el caso **b)**, en cambio, se da la situación de un patosistema[7] acuático, en donde la inoculación sucede a través del agua: los dos sitios amarillos son los más estrechamente conectados, a pesar de que están separados por una mayor distancia euclídea que con otros, porque el sitio

[4]Sistema de Información Geográfica.

[5]El grado en que el paisaje impide o facilita el movimiento entre las zonas de recursos [92].

[6]Introducción de microorganismos vivos, muertos o atenuados, en un organismo de forma accidental o voluntaria.

[7]Subsistema dentro del sistema agrícola caracterizado por el fenómeno de parasitismo. Está constituido por un hospedante susceptible, un patógeno virulento y un ambiente predispuesto a la enfermedad.

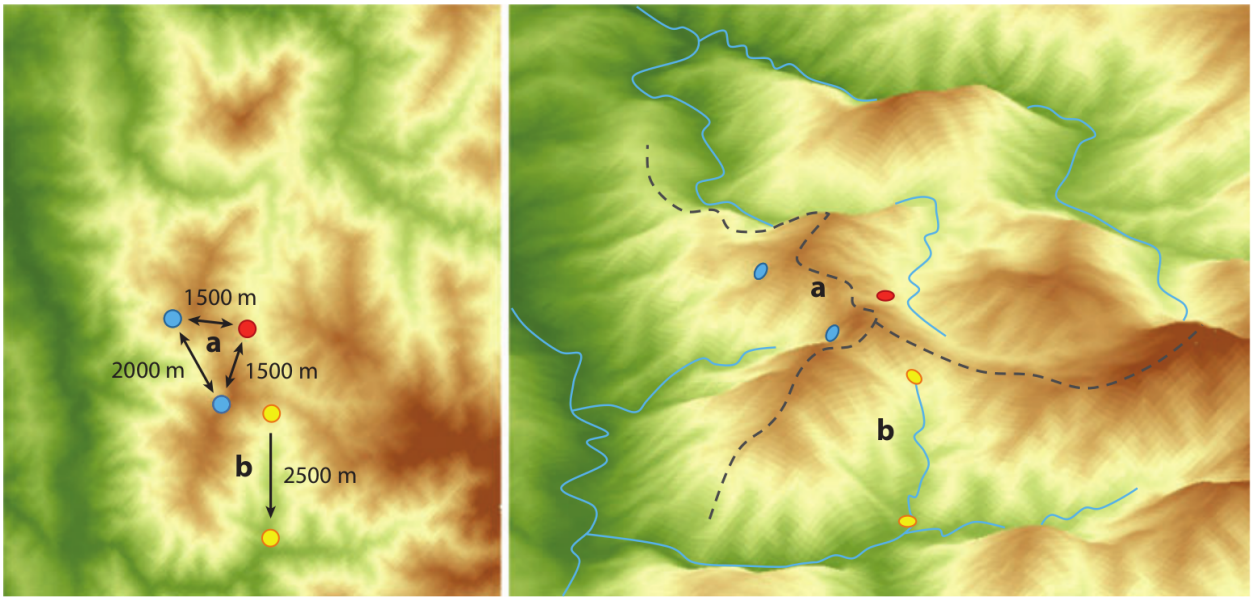


Figura 2-1.: La propagación y persistencia a través de paisajes heterogéneos.

amarillo de abajo está localizado bajo una corriente que va desde el sitio amarillo de arriba.

La ***Epidemiología Panorámica*** [64, 66] (EP) está estrechamente relacionada a su paralela ecológica, la Ecología Panorámica, una ciencia con inicios en los años 1930s que se dedica a estudiar las interacciones entre los ambientes y la vegetación. Sin embargo, los paisajes son espacial y temporalmente dinámicos. En simultáneo con el nacimiento de la ecología panorámica como una ciencia, Pavlosky estipula el concepto de *nidalidad*[8] (o focalidad) de las enfermedades, donde los patógenos son asociados a paisajes (zonas) específicos. Un foco de infección contiene tres elementos críticos [74]:

1. Vectores con capacidad de transmisión de la infección
2. Vertebrados capaces de funcionar como reservorio de la infección
3. Huéspedes susceptibles, como humanos o animales domésticos

El concepto de *focalidad* mezclado con la ecología panorámica llevó al nacimiento de la ciencia contemporánea ***Epidemiología Panorámica***, en la cual las enfermedades pueden ser asociadas a distintas características del paisaje o cómo la configuración entre el vector, el huésped y el patógeno se intersecan dado un clima permisivo para que ello suceda.

Por definición, la *EP* integra conceptos y enfoques de la ecología vinculada a las enfermedades, con el análisis a macroescala de la ecología del paisaje. La intersección de estas perspectivas nos habilita a entender cómo es que la configuración espacial y las características

[8] Se define como el foco de la infección. Además, Pavlosky, establece que los focos de enfermedades a microescala están determinados por todo el ecosistema [66].

de la composición del paisaje afectan a los procesos epidemiológicos a lo largo y ancho de las áreas geográficas que se extienden más allá de los procesos que operan localmente dentro de una sola comunidad. Así, la *EP* es más que simplemente establecer sectores en el territorio y examinar diferencias en condiciones locales de factores bióticos y abióticos entre distintos lugares. La clave es obtener información sobre la distribución geográfica de la enfermedad y comprender cómo las interrelaciones de los paisajes influencian las interreacciones entre individuos susceptibles e infectados.

La EP ha sido aplicada en gran variedad de estudios sobre vectores de enfermedades. A nivel global, se pueden encontrar contribuciones en esta área [8, 37, 43] también con algunas experiencias de herramientas operativas [6]. A su vez, muchos estudios interdisciplinarios fueron llevados a cabo en latinoamérica enfocados en la generación de modelos predictivos de riesgo, espaciales y temporales, basados en condiciones ambientales derivadas de información satelital [3, 25, 59, 65]. Por ejemplo, en México, Dumonteil y Gourbiere [32] estudiaron la relación entre la distribución de la especie Triatoma Dimidiata y factores bioclimáticos, para de esta forma desarrollar un modelo predictivo de la abundancia domiciliaria por esta especie y las tasas de infección por T. Cruzi. Estas predicciones se usaron para construir el primer mapa de riesgo de transmisión en la península de Yucatán hallándose que la abundancia de T. dimidiata se asocia de forma positiva (por análisis de regresión de Poisson) con los cultivos, pastos, precipitación, humedad relativa y la temperatura máxima. En particular, en Argentina existen varias experiencias en esta dirección. En [17, 21, 77] abordan el problema de la epidemia del Dengue dando herramientas operacionales [71].

Por ejemplo, en 2011, Argentina comenzó a desarrollar un proyecto operacional (Sistema de Alerta Temprana de Salud, HEWS), útil tanto para las autoridades de salud como para los investigadores. Básicamente, HEWS es un mapeo de riesgo dinámico del dengue para todas las ciudades del país. En este producto, cada ciudad es representada por un punto al que se le asigna un valor de riesgo para cada año, basado en tecnología geoespacial. El trabajo fue realizado en un contexto interdisciplinario e interinstitucional. En este sistema [71], el riesgo se evalúa en cuatro componentes que son: el entomológico, el viral, el componente relacionado con las actividades de control y finalmente el ambiental. Mientras que los tres primeros componentes se generan con el aporte de información de los agentes de salud que trabajan en cada ciudad, el cuarto se evalúa a partir de datos satelitales. Específicamente el componente ambiental, en la versión inicial del sistema, se evalúa con una probabilidad estacionaria de presencia de vectores (igual para todo el tiempo) más un componente relacionado con el número de ciclos virales, que son una función de la temperatura, diferente para cada ciudad y para cada año. El mapa de probabilidad de presencia de especie (modelo de nicho) es claramente una gran simplificación y se puede mejorar en base a datos satelitales continuos del medio ambiente. Variables como precipitación y temperatura, han demostrado, con una variabilidad local, influenciar el desarrollo de mosquitos, su supervivencia y actividad de oviposición[9] y por ende la abundancia de vectores.

[9]Proceso de implantación o difusión de huevos plenamente desarrollados a partir del cuerpo de la hembra.

Otro ejemplo a destacar en la utilización de éstas técnicas en Argentina es el trabajo de German y colaboradores [27], en 2017. En él desarrollan una metodología completa para generar modelos de manera automática y basada en información de libre acceso. En particular German [27] utiliza productos del sensor (MODIS) a bordo del satélite Terra y Aqua, pues es uno de los más adecuados para esta aplicación particular, debido a su resolución temporal, espectral y espacial. MODIS proporciona un conjunto de productos pre-procesados y de libre acceso [90]. Específicamente, los productos de vegetación (índice de vegetación de diferencia normalizada) y temperatura (temperatura de la superficie terrestre) derivados de MODIS son ejemplos de variables de percepción remota utilizadas en aplicaciones de epidemiologia [9, 71] incorporadas en [27]. Otra variable ambiental obtenida de satélite que es relevante e incorporada, es el *Índice de Agua de Diferencia Normalizada* (NDWI) que evalúa de alguna forma el contenido de agua de la cubierta terrestre. Adicionalmente el trabajo de German incorpora una estimación de la precipitación desde el espacio a partir de las misiones (TRMM) y (GPM) [48]. Utilizando los datos mencionados como variables independientes, desarrollaron modelos temporales de pronóstico de oviposición de *Aedes Aegypti* usando un método lineal multivariado. En su trabajo, mencionan que realizaron muchos métodos con distintos períodos de tiempo y se llegó a construir uno con una buena capacidad de predicción ($R^2 = 0{,}7$ utilizando 11 variables ambientales independientes en total)

2.2. Aprendizaje automático

Existen numerosos autores que han definido el concepto de que una máquina aprende. En este trabajo hemos extraído una en particular:

> *Se dice que un programa de computadora* ***aprende*** *de experiencia E con respecto a alguna tarea T y una métrica de rendimiento M, si con la experiencia E se incrementa su rendimiento en la tarea T, medida por M.*
>
> **Tom Mitchell, 1997** [56]

También, en el mismo libro, Mitchell enuncia que el campo del aprendizaje automático se refiere a la cuestión de cómo construir programas que mejoren automáticamente con experiencia. En ese marco, luego de muchos avances en el área, podemos decir que el *Aprendizaje Automático* (ML, por sus siglas en inglés) es un enfoque empírico efectivo para regresiones y/o clasificaciones de sistemas lineales y no lineales, que pueden involucrar desde unos pocos hasta varios cientos de variables.

Además, los métodos de ML se pueden clasificar en *supervisados* [12] y *no-supervisados* [33], aunque hoy en día existen matices entre estas dos clases [13]. Los algoritmos que aprenden a través de métodos supervisados son aquellos que aprenden una función que mapea un

valor de entrada a uno de salida basado en pares de ejemplos entrada-salida. Los algoritmos que utilizan métodos no-supervisados aprenden realizando inferencia de la función que describe la estructura de los datos de ejemplo. En este caso, los datos de entrenamiento del algoritmo no son etiquetados (no existen pares entrada-salida de ejemplo).

Los algoritmos bajo el enfoque de ML requieren entrenamiento utilizando un conjunto de datos que sea representativo del conjunto del problema. Además, para lograr modelos que puedan generalizar a datos nunca antes vistos, los algoritmos supervisados necesitan, al menos, dos subconjuntos necesariamente disjuntos de datos: el conjunto de entrenamiento y el de evaluación [82].

El ML es ideal para aquellos problemas en donde el conocimiento teórico del mismo es incompleto o insuficiente, pero se cuenta con un gran conjunto de observaciones. Este enfoque se utiliza, de manera creciente a medida que pasa el tiempo y el poder de cómputo se incrementa, en gran cantidad de aplicaciones tanto para problemas más relacionados al ámbito científico, como para problemas industriales. Algunos ejemplos de lo primero van desde problemas de procesamiento de lenguaje natural [10,63,72], procesamiento de imágenes [70,75,79] hasta aplicaciones en el área de la salud [31,60,89,97] y las Geociencias [11,49,83].

Estas técnicas han mostrado ser de utilidad para un gran número de aplicaciones en Geociencias relacionadas a la tierra, oceanos y atmósfera, y en algoritmos de extracción de información bio-geofísica. Algunos de los algoritmos de ML más usados en aplicaciones relativas a Geociencias y Sensado Remoto (GRS) son las Redes Neuronales Artificiales (ANN), Support Vector Machines (SVM), Mapas Auto-organizados (SOM), Árboles de Decisión (DT), Random Forests y Algoritmos Genéticos. Su aplicación en problemas de GRS es relativamente nuevo y extremádamente prometedor. En particular, ANNs son usadas para clasificación y la aplicación en pronósticos relativos a series de tiempo.

Una exploración en la base bibliográfica *Scopus* (`www.scopus.com`) devuelve más de 2.000 publicaciones que incluyen *remote sensing* y *machine learning* donde unas 900 fueron publicadas hasta el 2015, y alrededor de 1.200 desde ese año hasta la actualidad. Del total, el 24.5 % se corresponde con el área de *Computer Science*, un 21.3 % a *Earth and Planetary Sciences*, 16.5 % a *Engineering* y el restante 37.7 % se distribuye entre numerosas áreas. Esta búsqueda reflejó que China, Estados Unidos, Alemania e Italia son los países con mayor producción en este sentido.

A su vez, para tener una noción más exhaustiva sobre los esfuerzos académicos al respecto de los tópicos que se tratan en este trabajo, se realizó un búsqueda sobre la relación entre algunas de la herramientas que se han utilizado y el *remote sensing*.

Se encontró que si se cambian las palabras clave por *remote sensing* y *neural network* *Scopus* muestra que existen 4.000 publicaciones con esos tópicos, de las cuales alrededor de 1.500 fueron desde el 2015 hasta la actualidad. Del total, un 23.8 % se corresponde con el área de *Computer Science*, un 22.7 % a *Earth and Planetary Sciences*, 17.5 % a *Engineering* y el restante 36 % se distribuye entre otras áreas; con China, Estados Unidos, Italia e India como los paises con mayor producción científica en dichas áreas. El hecho de que esta búsqueda

haya arrojado más resultados que la mencionada anteriormente, indica que es posible que los investigadores que estén trabajando sobre problemáticas de este tipo, quizás, no están explotando las grandes capacidades del área de aprendizaje automático en su extensión y, en vez de eso, se están centrando en utilizar ANNs por su alta popularidad.

Otra exploración, esta vez sobre *remote sensing* y *k nearest*, arroja unos 1.100 resultados, de los cuales 310 fueron publicados desde el año 2015 a la actualidad. Esta vez, del total de publicaciones encontradas, un 25.7 % se corresponde con *Earth and Planetary Sciences*, 16.1 % a *Computer Science* y 15 % a *Engineering*. Para este caso, Estados Unidos lidera fuertemente la lista de paises con más producción, con 387 trabajos. China y Alemania lo siguen con 320 publicaciones entre los dos.

A continuación, expondremos de manera más detallada herramientas y métodos que motivaron el desarrollo del presente trabajo. Enfocaremos en los métodos de regresión, dado que es ésta la clase de problema que abordaremos. A su vez, los algoritmos que describiremos son los implementados por la librería *Scikit-learn* [67].

2.2.1. Métodos Lineales

En este trabajo utilizamos dos tipos de regresiones lineales. La regresión lineal ordinaria, correspondiente al método de *Mínimos Cuadrados* [28] y método de regresión *Ridge* [40]. Dada su popularidad y simplicidad, no ahondaremos en explicaciones profundas con el fin de evitar detalles tediosos, muy conocidos.

2.2.2. Árboles de Decisión

Los *Árboles de Decisión* (DTs, por sus siglas en inglés) [96] son métodos no paramétricos de aprendizaje supervisado utilizados tanto para problemas de clasificación como de regresión. La meta es crear un modelo que prediga el valor de una variable objetivo aprendiendo simples reglas de decisión inferidas a partir de las características de los datos de entrenamiento.

El algoritmo de aprendizaje de los DT construye modelos de clasificación o regresión utilizando una estructura arbórea. Éste divide el conjunto de datos en pequeños subconjuntos mientras que, al mismo tiempo, un árbol de decisión es incrementalmente construido. El resultado final es un árbol con nodos de decisión y nodos hojas. Un nodo de decisión tiene dos o más ramas, cada una representando valores para el atributo examinado. Un nodo hoja representa una decisión dentro del objetivo numérico. Los árboles de decisión pueden manejar tanto datos categóricos como numéricos.

Más formalmente, dados vectores de entrenamiento $x_i \in \mathbb{R}^n$, $i = 1, .., l$ y un vector de etiquetas $y \in \mathbb{R}^l$, un árbol de decisión particiona recursivamente el espacio de modo que las muestras con la misma etiqueta se agrupen juntas.

Supongamos que los datos en el nodo m son representados por el conjunto Q. Para cada

candidato se divide $\theta = (j, t_m)$ donde j es una característica y t_m es un umbral, particionando los datos en conjuntos $Q_{izq}(\theta)$ y $Q_{der}(\theta)$ donde:

$$Q_{izq}(\theta) = (x, y)|x_j \leq t_m \tag{2-1}$$
$$Q_{der}(\theta) = Q - Q_{izq}(\theta) \tag{2-2}$$

La impureza en m es calculada usando la función de impureza H. La elección de ésta depende de la tarea que se quiera realizar. En el caso de una regresión, los criterios para minimizar en cuanto a la determinación de ubicaciones para las divisiones suelen ser el **Error Cuadrático Medio**, que minimiza el error $L2$ [62] usando valores promedios en los nodos terminales, y el **Error Absoluto Medio**, que minimiza el error $L1$ [62] usando el valor de la mediana estadística en los nodos terminales. Y así, una vez seleccionada la función de impureza, para el nodo m, representando una región R_m con una cantidad N_m observaciones se define la función G de la siguiente manera:

$$G(Q, \theta) = \frac{n_{izq}}{N_m} \; H(Q_{izq}(\theta)) + \frac{n_{der}}{N_m} \; H(Q_{der}(\theta)) \tag{2-3}$$

y se seleccionan los parámetros que minimicen la impureza tal como expresa la siguiente ecuación

$$\theta^* = argmin_\theta \; G(Q, \theta) \tag{2-4}$$

Luego se sigue partiendo Q_{izq} y Q_{der} hasta que se alcance la profundidad máxima permitida del árbol, $N_m < min_{muestras}$ o bien $N_m = 1$.

2.2.3. Random Forest[10]

Random Forest (RF) es un método de aprendizaje que utiliza ensamblado de DTs y se usa para llevar a cabo tareas tanto de clasificación como de regresión. La idea es construir una variedad de árboles de decisión en tiempo de entrenamiento y devolver la clase que se corresponda con la moda estadística de las clases (para clasificación) o bien el promedio (para regresión) de los resultados obtenidos por los árboles individuales.

Existen varios algoritmos de RF. Describiremos formalmente el desarrollado por Breiman [7]. Sea $\mathcal{D}_n = \{(X_1, Y_1), \ldots, (X_n, Y_n)\}$ un conjunto de variables aleatorias independientes e idénticamente distribuídas (i.i.d.) pertenecientes al conjunto $[0, 1]^d \times \mathbb{R}$ con $d \geq 2$, con la misma distribución que un par genérico (X, Y) satisfaciendo que $\mathbb{E}Y^2 < \infty$. Además, sea $r()$ la función de regresión que se busca estimar.

Un RF es un predictor que consiste de una colección aleatoria base de árboles de regresión, $\{r_n(x, \theta_m, \mathcal{D}_n), m > 1\}$, donde $\theta_1, \theta_2, \ldots$ son salidas i.i.d. de una variable aleatoria θ. Éstos árboles aleatorios son combinados para formar la estimación de la regresión:

[10]Dentro de la comunidad técnica, algunos algoritmos se referencian utilizando los términos en inglés aunque el idioma en el cual se expresen sea el castellano.

$$\overline{r}(X,\ \mathcal{D}_n)\ =\ \mathbb{E}_\theta[r_n(X,\ \theta,\ \mathcal{D}_n)] \tag{2-5}$$

donde $\mathbb{E}_\theta$ denota la esperanza con respecto al parámetro aleatorio condicionada por X y el conjunto de datos $\mathcal{D}_n$. Notemos que dicha esperanza es evualuada por el método de Monte Carlo [54], esto es, generando M árboles aleatorios, y tomando el promedio de los resultados. La variable de aleatoriedad θ es usada para determinar cómo se van a realizar los sucesivos cortes cuando se construyen los árboles individuales, como una selección de la coordenada a dividir y la posición de la división.

Cada árbol aleatorio es construido de la siguiente manera: todos los nodos del árbol son asociados a celdas rectangulares tales que en cada etapa de construcción del árbol, el conjunto de celdas asociadas a las hojas del árbol forman una partición de $[0,1]^d$. La raíz del árbol es exactamente $[0,1]^d$. Luego, el siguiente procedimiento es repetido una cantidad $\lceil \log_2 k_n \rceil$ veces donde k_n es un parámetro determinístico, fijado por el usuario, posiblemente dependiente del valor de n.

1. En cada nodo, se selecciona una coordenada de $X = (X^{(1)},\ \ldots,\ X^{(d)})$ donde la característica j-ésima tiene una probabilidad de $p_{nj} \in (0,1)$ de ser elegida.

2. En cada nodo, una vez que la coordenada es seleccionada, la división es en el punto intermedio del lado elegido.

Cada árbol aleatorio $r_n(X,\theta)$ devuelve el promedio sobre todos los Y_i para los cuales los vectores correspondientes X_i caen en la misma celda de la partición aleatoria que X.

2.2.4. K-Vecinos más cercanos (KNN)

El principio detrás de los métodos de vecinos más cercanos es encontrar un número predefinido de las muestras de entrenamiento más cercanas en distancia al nuevo punto, y predecir su valor a partir de ellos. El número de muestras puede ser una constante definida por el usuario, o variar basada en la densidad local de los puntos. La distancia puede, en general, ser cualquier métrica aunque la distancia Euclídea es la elección más común.

Las predicciones son hechas para un nuevo punto x, buscando a través del conjunto de entrenamiento completo las K instancias más cercanas (los vecinos) y computar la variable de retorno utilizando la información de esos K puntos. Para el caso de la regresión suele ser el promedio de cada variable de retorno de la siguiente manera

$$\overline{y}(x) = \frac{1}{k} * \sum_{j \in knn(x)} y_j \tag{2-6}$$

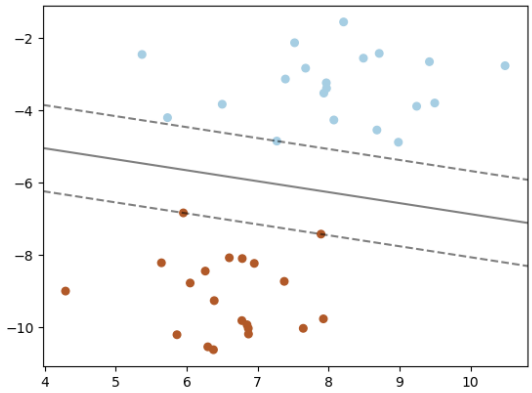


Figura 2-2.: Plano de separación de clases generado por una SVM.

2.2.5. Support Vector Machine (SVM) y Support Vector Regression (SVR)

Una *Support Vector Machine* (SVM) [5] construye un hiperplano o un conjunto de hiperplanos en un espacio de muy alta, o infinta, dimensionalidad. Éstos pueden ser usados tanto para tareas de regresión como de clasificación. Intuitivamente, una buena separación se consigue por el hiperplano que tenga la mayor distancia a los puntos de entrenamiento más cercanos de alguna clase, como se puede observar en la Figura **2-2**, dado que en general a más grande sea esa distancia más pequeño será el error de generalización del modelo.

Support Vector Regression (SVR) [4, 15] es una veloz y precisa forma de interpolación de conjuntos de datos. Es útil cuando se quiere aproximar una función costosa de calcular sobre un dominio conocido. Aprende rápidamente y se puede mejorar sistemáticamente.

SVR es una generalización de la SVM a problemas de regresión. Técnicamente, se puede decir que es un algoritmo de aprendizaje supervisado. Éste requiere de un conjunto de datos de entrenamiento, $\mathcal{T} = (\vec{X}, \vec{Y})$, que cubre el dominio de interés acompañado de las soluciones en dicho dominio. El trabajo de la SVM es aproximar la función definida por el conjnuto de entrenamiento, $F(\vec{X}) = \vec{Y}$. En general, en las SVM, los vectores $\vec{X}$ son utilizados para definir el hiperplano que separa las distintas soluciones posibles. En problemas de regresión, estos vectores son utilizados para realizar una regresión lineal. Los que estén más cerca del punto de prueba se los llama *vectores de soporte*.

Daremos una idea más detallada de los fundamentos matemáticos detrás de una SVR [88]: Dados los vectores de entrenamiento $x_i \in \mathbb{R}^p$ con $i = 1, \ldots, n$ y un vector $y \in \mathbb{R}^n$, el ϵ-SVR resuelve el siguiente problema primario:

$$\min_{w,b,\zeta,\zeta^*} \frac{1}{2} w^T w + C \sum_{i=1}^{n} \zeta_i \tag{2-7}$$

con las restricciones de:
$y_i - w^T \phi(x_i) - b \leq \epsilon + \zeta_i$,
$w^T \phi(x_i) + b - y_i \leq \epsilon + \zeta_i^*$,
$\zeta_i^*, \zeta_i \geq 0$
para $i = 1, \ldots, n$

Mientras que el problema dual a resolver es:

$$\text{mín}\, \alpha, \alpha^* \frac{1}{2} (\alpha - \alpha^*)^T Q (\alpha - \alpha^*) + \epsilon e^T (\alpha + \alpha^*) - y^T (\alpha - \alpha^*) \tag{2-8}$$

con las restricciones de:
$e^T (\alpha - \alpha^*) = 0$,
$0 \geq \alpha, \alpha^* \leq C$
para $i = 1, \ldots, n$. Donde e es un vector para el cual todos sus componentes poseen el valor 1, $C > 0$ es la cota superior, Q es una matriz semidefinida positiva[11] de tamaño $n \times n$, $Q_{ij} \equiv K(x_i, x_j) = \phi(x_i^T)\phi(x_j)$ es el núcleo (*kernel*, en inglés). Aquí, los vectores de entrenamiento están siendo mapeados a un espacio de gran (probablemente infinita) dimensionalidad por la función ϕ. Luego, la función de decisión es:

$$\sum_{i=1}^{n} (\alpha - \alpha^*) K(x_i, x) + \rho \tag{2-9}$$

2.2.6. Perceptron Multicapa (MLP)

Un **Perceptron Multicapa** (MLP, por sus siglas en inglés) [61, 80] es un tipo de red neuronal artificial (ANN) *feedforward*. Es un algoritmo de aprendizaje supervisado que logra distinguir relaciones entre datos que no sean linealmente separables. Aprende una función a partir de un conjunto de datos de entrenamiento y puede ser utilizada tanto para tareas de regresión como de clasificación haciendo uso, entre otras cosas, de una técnica llamada *propagación hacia atrás* [36] (*backpropagation*, en inglés). El MLP consiste de al menos tres capas: una de entrada, una de salida y, como mínimo, una capa oculta[12]; la vista gráfica de una arquitectura simple se puede observar en la Figura **2-3**.

Una descripción matemática simplificada del algoritmo en cuestión es la que se menciona a continuación. Dados ejemplos de entrenamiento $(x_1, y_1), \ldots, (x_n, y_n)$ donde $x_i \in \mathbb{R}^m$ y $y_i \in \{0, 1\}$, un MLP de una capa oculta con una neurona aprende una función $f(x) = W_2 g(W_1^T x + b_1) + b_2$ donde $W_1 \in \mathbb{R}^m$ y $W_2, b_1, b_2 \in \mathbb{R}$ son parámetros del modelo. W_1, W_2 representan

[11] Una matriz, M, es semidefinida positiva si $x^* M x \leq 0 \; \forall x \in \mathbb{R}^n$.

[12] Una capa de neuronas artificiales que toman un conjunto de entradas ponderadas y producen una salida a través de una función de activación.

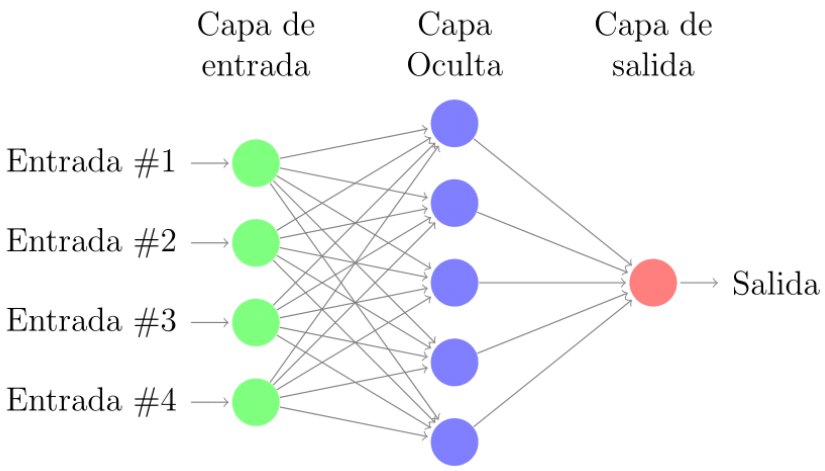


Figura 2-3.: Arquitectura de una MLP de cuatro variables de entrada, una capa oculta de cinco neuronas y un sólo valor de salida.

los pesos de la capa de entrada y la capa oculta, respectivamente. b_1, b_2 representan el sesgo agregado a la capa oculta y la capa de salida, respectivamente. La función $g : \mathbb{R} \to \mathbb{R}$ es la función de activación, la cual es definida por defecto como la tangente hiperbólica. Está dada por,

$$g(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}} \tag{2-10}$$

En problemas de regresión, la salida del algoritmo es $f(x)$, por lo que la función de activación de salida es simplemente la función identidad. En estos problemas, MLP también utiliza como función de pérdida la correspondiente al *Error Cuadrático*:

$$Loss(\hat{y}, y, W) = \frac{1}{2} \|\hat{y} - y\|_2^2 + \frac{\alpha}{2} \|W\|_2^2 \tag{2-11}$$

Comenzando desde pesos con valores aleatorios, el MLP minimiza la función de pérdida actualizando repetidamente dichos pesos. Luego de calcular la pérdida, se propaga desde la capa de salida a todas las anteriores (*backpropagation*), proporcionando un valor de peso a cada parámetros para disminuir la pérdida. Para ello, se utiliza *descenso por gradiente*, en el cual el gradiente $\nabla Loss_W$ de la pérdida con respecto a los pesos es calculada y deducida de W. Más formalmente, es expresado como:

$$W^{i+1} = W^i - \epsilon \nabla Loss_W^i \tag{2-12}$$

donde i es el paso de iteración, y $\epsilon > 0$ es la taza de aprendizaje.

En general el algoritmo termina cuando se alcanza cierto número definido por el usuario de iteraciones o se cruza un umbral para la pérdida.

3. Modelando la población del vector de Dengue utilizando datos de sensado remoto y aprendizaje automático

En un contexto interinstitutional entre la Comisión Nacional de Actividades Espaciales (CONAE) y el ministerio de salud de Argentina, se han desarrollado iniciativas orientadas a modelar la evolución temporal de las poblaciones de mosquitos usando variables ambientales obtenidas de sensores remotos. Estos trabajos utilizaron series de algunos años y fueron basadas en un pequeño número de variables satelitales [18, 22]. En un esfuerzo para mejorar esto, se construyeron modelos de series temporales de cuatro años, basados en una gran cantidad de variables de varios sensores [16]. Aún así, todos estos trabajos asumieron modelos lineales multivariados.

Como parte del trabajo presentado en este capítulo se desarrolló un sistema para la generación y evaluación de modelos basados en aprendizaje automático. Ésto brinda a la comunidad de profesionales que aborda las problemáticas relacionadas con la epidemiología, una herramienta de alto valor dado que les permite realizar modelos de evolución de población de mosquitos.

A su vez, este trabajo representa una mejora en la capacidad de modelado con respecto a los esfuerzos previos ya mencionados. En éste se comparan distintos algoritmos de aprendizaje automático: Support Vector Machines (SVMs), Redes Neuronales Artificiales (ANNs), K-vecinos más cercanos (KNNs) y un tipo de árbol de decisión orientado a regresión. Se suman a la comparación dos modelos de regresión lineal. Con ésto, se obtiene una metodología operacional que podría contribuir al sistema de riesgo de Dengue actualmente en operación [71, 78].

Adicionalmente, se explora, en contraste con los trabajos previos mencionados, la habilidad de modelado y predicción de oviposición con algoritmos de aprendizaje automático *off-the-shelf*, i.e. algoritmos de software libre, ya implementados, sin mayores desarrollos sobre lo existente y con un ajuste de hiperparámetros mínimo. De ésta manera se busca la asimilación de estas técnicas a toda la comunidad que se ocupa de problemas similares.

Finalmente, resulta relevante mencionar que el trabajo realizado y descripto aquí ha dado lugar a la publicación *Modeling Dengue Vector Population Using Remotely Sensed Data and Machine Learning* [86] en la revista *Acta Tropica* de *Elsevier*[1].

[1] `https://www.journals.elsevier.com/acta-tropica`

3.1. Obtención, análisis y selección de datos a utilizar

3.1.1. Datos de estudio y Datos de Campo

Figura 3-1.: Área de Estudio

El estudio presentado fue desarrollado en la ciudad de Tartagal (con 79.900 habitantes) en el noroeste de Argentina (22°32′ S, 63°49′ O, 450 m sobre el nivel del mar), en la provincia de Salta. El sitio está entre 50 y 100 kms de la frontera entre Argentina y Bolivia, como se puede apreciar en la Figura **3-1**.

Este lugar tiene una temperatura media anual de unos 23 °C (máximo promedio de verano de 39 °C y mínimo promedio en invieron de 9 °C). Tiene una precipitación anual de 1100 mm, con una estación seca (Junio a Octubre). Tartagal, como muchas ciudades del noroeste argentino, tiene una diversidad cultural basada en la presencia de grupos étnicos autóctonos y población de inmigrantes sumada al movimiento de migración proveniente de Bolivia. Estas características conducen a un perfil peculiar de comportamiento cultural, social y economico.

La población de vectores es medida monitoreando la actividad de oviposición. Para ello se utilizan ovitrampas colocadas en casas aleatoriamente seleccionadas en el área urbana de la ciudad. El período de monitoreo utilizado en este estudio fue de Agosto de 2012 hasta Julio de 2016 sobre 50 casas. Dos ovitrampas fueron colocadas en cada una: una dentro y otra fuera, en el patio trasero en un lugar con sombra y a nivel del suelo, siguiendo las instrucciones de la OMS [57]. Las ovitrampas son contenedores de 1000 cm^3 de plástico negro con 250 mL de agua sin ninguna infusión de atracción. En este estudio sólo utilizamos los datos de las ovitrampas externas dado que ellas tienen una mayor correlación con las variables ambientales derivadas de información satelital. Dichas ovitrampas son reemplazadas semanalmente y los huevos son contados en un laboratorio de acuerdo al *Indice de Densidad de Huevos* [29]. Luego, la

actividad de oviposición del *Aedes Aegipty* es estimada por la suma de los huevos capturados en las trampas externas de la ciudad.

3.1.2. De productos satelitales a variables ambientales: conjunto de datos para el modelado

Siguiendo la idea de construir modelos predictivos de la población de vectores basados en variables ambientales derivadas de satelites, pero con una perspectiva operacional basada en trabajos previos, se generaron representaciones de vegetación, humedad, temperatura y lluvia operacionalmente disponibles de **MODIS** y productos **TRMM/GPM**.

Los índices de vegetación global proveen productos espaciales y temporales consistentes sobre la cobertura de la vegetación, propiedades del área foliar y el nivel de clorofila. Estos indices son derivados de la reflectancia atmosférica corregida en las bandas infrarroja media (MIR, por sus siglas en inglés) y cercana (NIR, por sus siglas en inglés). En este trabajo se utiliza el **NDVI** del producto satelital de MODIS, *MOD13Q1*, (compuesto de 16 días) con una resolución espacial de 250 m. Las condiciones de vegetación son incluidas junto con la temperatura, humedad y precipitación, las cuales son variables relevantes para la evolución de la población de mosquitos [22, 35].

A su vez, se incluye el Indice de Agua de Diferencia Normalizada (**NDWI**), que está vinculado al contenido de agua líquida y humedad tanto en la vegetación como en estructuras sólidas. Es calculada a partir del mismo producto MODIS usando la definición de *Gao* [26] del NDWI desde las bandas provistas por el producto *MOD13Q1*, correspondiente a la reflectancia de MIR y NIR: $NDWI = (\rho_{NIR} - \rho_{MIR})/(\rho_{NIR} + \rho_{MIR}) \times 10^4$. Los productos MODIS, en general, necesitan el factor 10^4 para ser guardados, por eficiencia computacional, como números enteros.

Por su parte, utilizamos la temperatura de la superficie terrestre (**LST**) de MODIS dado que es una aproximación de la temperatura ambiental [44, 68, 94]. Para esto, se eligió el producto satelital *MOD11A2*. Éste tiene una resolución espacial de 1 km y es un promedio de valores de LST de cielo-abierto durante un periodo de 8 días. Este producto incluye LST de la noche y del día para así, de alguna manera, representar temperaturas mínimas y máximas [95].

Finalmente, la precipitación local es obtenida de la *Misión Tropical de Medida de Lluvia* (**TRMM**) [48]. Ésta es una misión conjunta entre la NASA y la Agencia Aeroespacial de Exploración de Japón lanzada en 1997 para el estudio de las lluvias y así realizar investigaciones sobre el clima. Para detectar la lluvia, el satélite utiliza muchos instrumentos incluyendo radar, imágenes de microondas y sensores de rayos. TRMM, a pesar de que se quedó sin combustible en 2014, siguió transmitiendo datos hasta Junio del 2015. Luego de eso, otros productos basados en una nueva misión espacial llamada GPM[2], fueron publicados para asegurar la continuidad de estos trabajos.

[2] `https://earthdata.nasa.gov/trmm-to-gpm`

Dos áreas de 85 ha fueron definidas alrededor de la ciudad de Tartagal y se calcularon los valores medios para todas las variables derivadas de satelite. Siguiendo el enfoque de [16, 19, 20], la primer área se encuentra ubicada dentro de la ciudad (Área Urbana) y la segunda abarca la vegetacion nativa que rodea la ciudad (Área Rural). Ésta elección fue tomada bajo la hipótesis de que seleccionar una zona fuera de la ciudad representaría bien las condiciones ambientales (NDWI, NDVI, LST). La misma idea fue utilizada en estudios previos muy relacionados. Como se explica en dichos trabajos previos, es esperable que estas observaciones y los índices de larvas estén estrechamente relacionadas. En ese sentido es que el área rural o externa fue seleccionada aleatoriamente de aquellas con vegetación suficientemente cercana a la ciudad, representando condiciones ambientales naturales. En este caso específico esta región rural es seleccionada en el noreste de la ciudad; tiene una altitud similar a la ciudad y mayormente bosque nativo. Se puede observar en la Figura **3-2**.

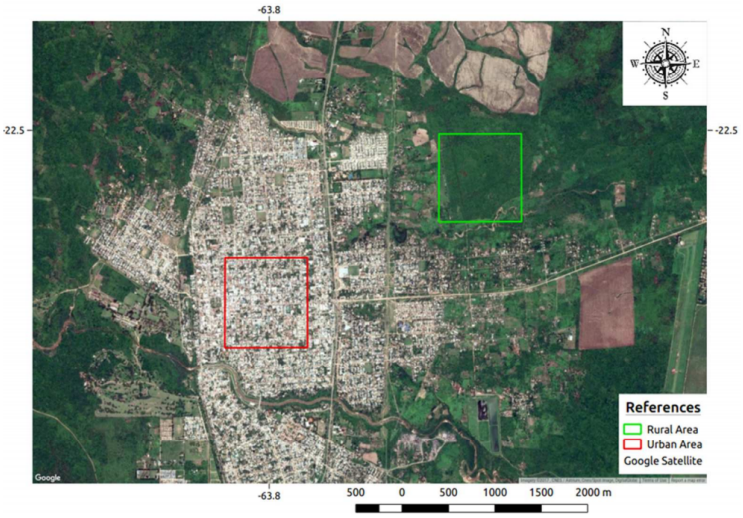


Figura 3-2.: Areas rural y urbana seleccionadas para extraer las variables ambientales.

El procedimiento de construir las series temporales a partir de variables provenientes del sensado remoto se describen en la Figura **3-3**. Las imágenes son obtenidas de la NASA[3] e importadas en **GRASS 7.1**. Para cada una de las áreas anteriormente definidas se calcula la media de cada día. Cada uno de estos valores promedio y sus fechas son exportadas a una tabla en el software **R** [73], donde es utilizada para construir las series temporales completas. Los datos son interpolados para obtener valores para cada uno de los días de muestra (un valor para cada semana epidemiológica) [27].

Todas las variables son consideradas con hasta tres semanas de *lag*[4] teniendo en cuenta

[3] `http://e4ftl01.cr.usgs.gov`

[4] Para un punto en el tiempo t, el valor de la variable $v_{1lag_1}(t)$ es igual al valor de la variable $v_1(t-1)$ correspondiente al punto en el tiempo de una semana anterior.

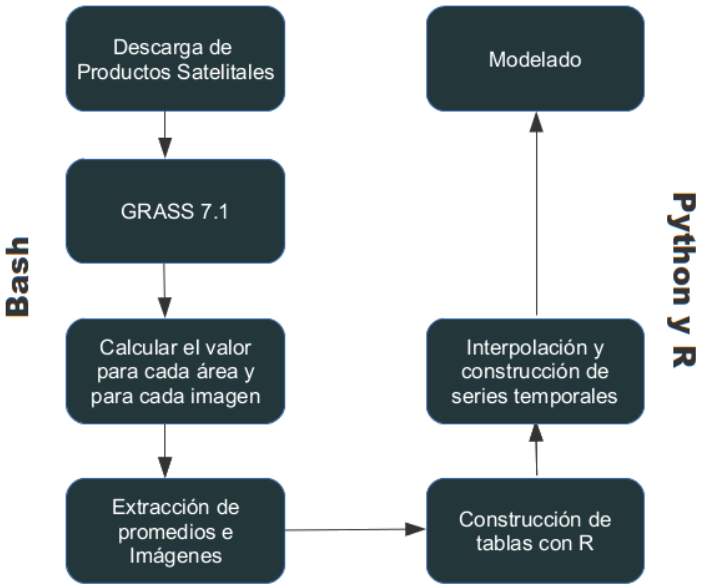


Figura 3-3.: Sistema de procesamiento de productos satelitales.

las series temporales originales, para representar las influencias asincrónicas, correspondientemente con uno, dos o tres lapsos de tiempo.

El primer paso consistió en analizar las cuarenta variables ambientales y los huevos recolectados cada semana por medio de una matriz de correlación y los valores **p** que miden su significancia. Ésto llevó a descartar treinta y cinco variables. Se prefieren las variables con *lag* dada su potencial habilidad de pronóstico [27]. Las siguiente variables fueron seleccionadas: NDVI rural *lag* 1, NDWI rural *lag* 1, LST rural dia *lag* 3, LST rural noche *lag* 1 y TRMM *lag* 3. Luego, todas las variables fueron normalizadas utilizando el ***z-score***[5].

La Figura **3-4** presenta las variables ambientales junto con la oviposición en un mapa de calor (*heatmap*). Este formato permite una visualización en la evolución temporal, de los patrones de correlación entre las variables y el efecto de *lag*.

3.2. Modelado

Con el conjunto de datos descripto en la sección anterior, se implementaron dos modelos lineales (tradicional y *Ridge*) y cuatro modelos no-lineales (*Support Vector Machine*, ANN Perceptron Multicapa, Árbol de Decisión, K-vecinos más cercanos) para modelar la oviposición para cada semana. Para todos los modelos se utilizó el mismo conjunto de 5 variables ambientales como *features*.

[5]Técnica para normalizar datos en función de la media y desviación estándar muestrales.

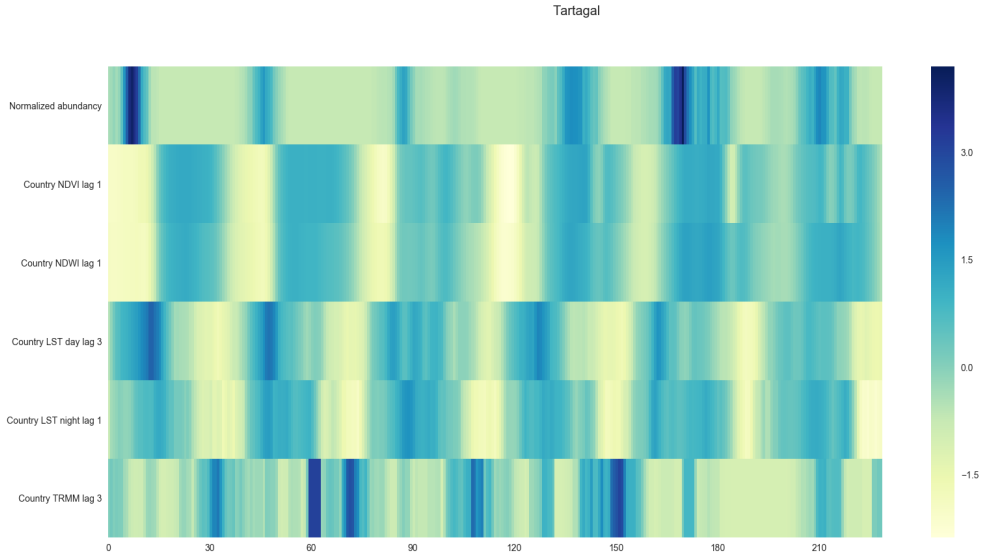


Figura 3-4.: Heatmap de valores del *z-score* de variables ambientales a través de las semanas para las cuales se poseen datos.

En todos los casos se entrenaron los modelos con 80 % del conjunto de datos y se utilizó el 20 % restante de la serie temporal (alrededor de un año) como un conjunto independiente para validar la capacidad de predicción temporal de las herramientas (se usó el 20 % más "nuevo"del conjunto). Esta elección de porcentajes de división es la más utilizada en la literatura de ML [14].

Se utilizó **Validación Cruzada** (*Cross Validation*) [14, 69] con el objetivo de reducir la dependencia de los resultados en una selección particular del par de conjuntos de entrenamiento y validación. En particular, para evaluar los modelos, se utilizó un procedimiento de validación cruzada particular para problemas que involucran series temporales[6]. Otras técnicas de validación cruzada como *K-fold* no son adecuadas para los datos que se corresponden con series temporales, i.e, cuando el orden en el conjunto de datos es importante.

A continuación se describirán las técnicas utilizadas para modelar el *z-score* de la oviposición como una función variables ambientales extraidas de sensores remotos. Todos los modelos fueron implementados utilizando funciones de la librería ***scikit-learn***. Es una librería del lenguaje de programación Python, de libre acceso, para Aprendizaje Automático contruida sobre *SciPy* [42]. Es una herramienta sencilla y efectiva para minería y análisis de datos que proporciona un conjunto de utilidades que permiten una implementación completa de la solución de un problema de ML. Dado que está bajo licencia *BSD*, esta librería puede ser utilizada tanto para uso personal como comercial.

[6] `http://scikit-learn.org/stable/modules/cross_validation.html`

3.2.1. Sistema de Modelado

Requerimientos

Dado el objetivo de este trabajo, los requerimientos del mismo se basaron en la compatibilidad con lo publicado en el artículo "*An operative dengue risk stratification system in argentina based on geospatial technology*" [71]. Luego de un análisis del mismo, se concluyó que el sistema de modelado debe poseer las siguientes características:

- Facilidad de utilización para un usuario no especialista del área de Ciencias de la Computación.
- Poseer una herramienta de limpieza del conjunto de datos dado, para construir automáticamente los datos que luego utilizarán los distintos algoritmos.
- Versatilidad para su uso con otros conjuntos de datos sin necesidad de realizar mayores cambios en la arquitectura del sistema.
- Debe poseer una herramienta para la generación de instancias de datos para el ajuste de hiperparámetros y conjuntos de entrenamiento y validación de los modelos.
- Generar modelos que queden dispuestos para su evaluación en nuevos datos.
- Dichos modelos deben ser serializados para facilitar la puesta en operatividad.
- Versatilidad para agregar nuevos modelos. Éstos deben poseer funciones `fit` y `predict` para entrenar el modelo y realizar inferencias, correspondientemente.
- Debe poseer una herramienta que permita evaluar la capacidad de predicción de un modelo a través de gráficos.
- Debe poseer una herramienta de relativa sencillez de utilización para el ajuste de hiperparámetros de los modelos.

A su vez, el desarrollo se llevó a cabo utilizando una metodología en cascada. Ésto quiere decir que primero se establecieron los requerimientos, luego se realizó una investigación de las herramientas que potencialmente se utilizarían para generar un diseño y establecer la arquitectura del proyecto. Posteriormente se codificó y se realizaron las pruebas de usuario necesarias.

Arquitectura

Como se ve en la Figura **3-5** el sistema de modelado tiene tres módulos importantes: `data`, `models` y `tunning`.

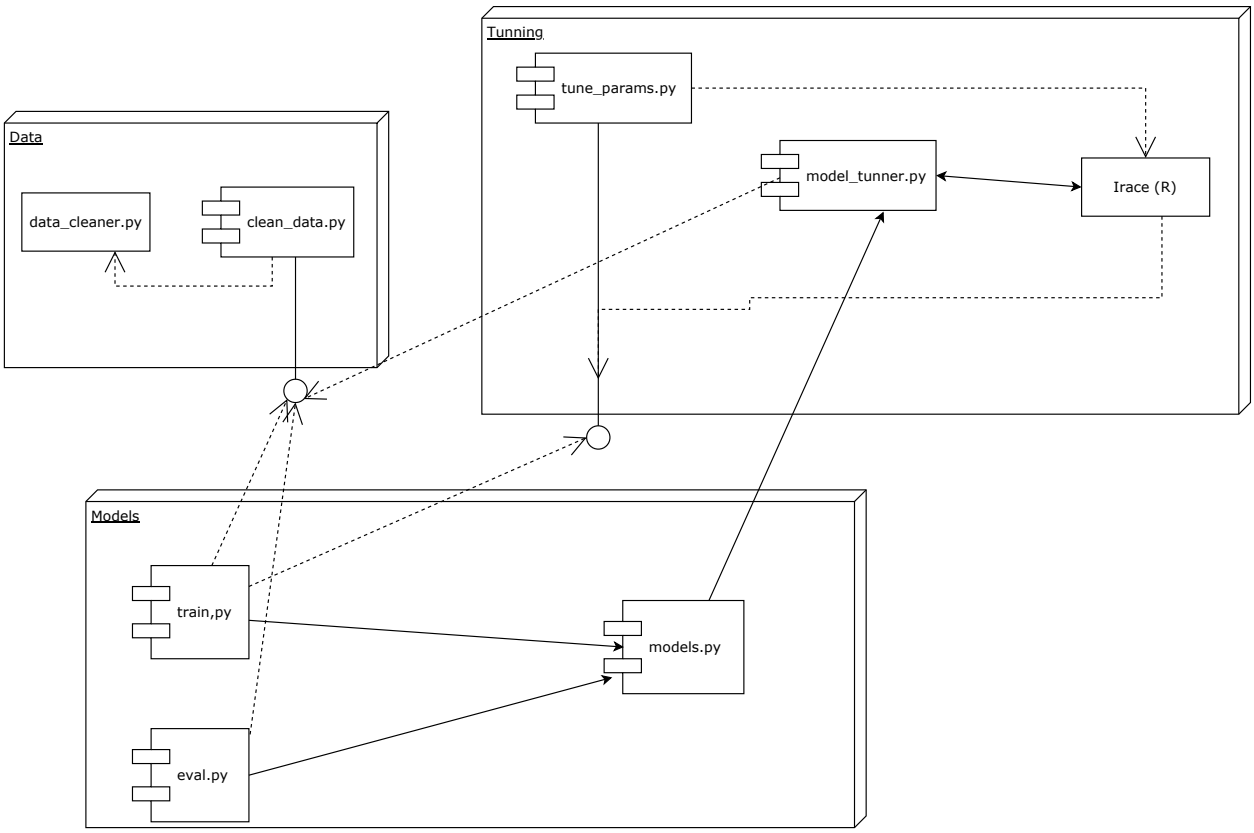


Figura 3-5.: Sistema para el ajuste de parámetros y modelado.

En el módulo `data`, como se puede deducir de su nombre, es el encargado de limpiar los conjuntos de datos y generar los bloques de entrenamiento, validación y las instancias para realizar el ajuste de hiperparámetros de cada algoritmo.

`models` es el módulo dedicado a definir los algoritmos y posee los *scripts* para el entrenamiento y evaluación de los mismos. Sumado a ésto, en este modulo se encuentra el archivo donde se deben colocar los modelos que serán utilizados en el modelado.

`tunning` es el módulo encargado del ajuste de hiperparámetros de los modelos. Éste realizará una búsqueda sobre el espacio de parámetros para encontrar los óptimos para cada algoritmo. Este procedimiento utiliza la herramienta *irace* del lenguaje *R* desde una interfaz de *Python*.

Más detalles de código se encuentran en el Anexo A.

3.2.2. Modelos lineales

Experiencias previas en aplicaciones epidemiológicas de modelado utilizando variables ambientales obtenidas de sensores remotos han reportado buenos resultados con este enfoque [2, 18, 47]. En este caso se utilizó un modelo de regresión lineal tradicional y una regresión *Ridge*, esta última con la regularización de *Tikhonov* y validación cruzada. Cabe destacar

que la regresión *Ridge*, en la literatura de ML, se la suele denominar como "decaimiento de peso"(*weight decay* [46]).

3.2.3. Modelos no-lineales

A diferencia de los modelos lineales, los no-lineales son capaces de capturar relaciones funcionales más complejas entre los datos, con el costo de una complejidad computacional más grande y una carga mucho mayor para el usuario que debe realizar un trabajo más fino de ajuste del modelo (la selección de hiperparámetros, entre otras).

Tipicamente, una regresión en el ámbito del aprendizaje automático incluye cuatro pasos fundamentales:

- Análisis del conjunto de datos: implica extraer variables de interés, quitar redundacia y valores que generen ruido, etc
- Arquitectura: Selección del algoritmo y de los hiperparámetros como la cantidad de capas y neuronas en una ANN, el número de vecinos en el algoritmo de K-vecinos más cercanos, etc.
- La etapa de entrenamiento-validación: los parámetros del modelo se ajustan y se realizan técnicas de validación de dicho modelo para medir el desempeño del modelo para generalizar a nuevos datos.
- Utilizar el modelo con datos nuevos.

Estos pasos se implementaron utiliizando, mayormente, funciones disponibles en la librería *scikit-learn* ya mencionada.

La configuración o selección del conjunto óptimo de hiperparámetros en este tipo de modelos no-lineales es un problema complejo. Ésto podría realizarse a mano o utilizando herramientas semi automáticas. Lo primero no es buena práctica dado que podría generar un sesgo sobre los valores obtenidos y, además, el gran número de posibles combinaciones requeriría mucho tiempo del usuario en ésta tarea, aún así hay ocasiones en las que se utiliza esta metodología.

Para realizar el ajuste de hiperparámetros, en este trabajo utilizó el paquete ***IRace*** (*Iterated Racing for Automatic Algorithm Configuration*) [51]. Ésta herramienta realiza un procedimiento iterativo capaz de encontrar automáticamente la configuración de hiperparámetros más apropiada dadas las instancias de datos generadas para esta etapa. Está disponible gratuitamente para el lenguaje **R** en `http://iridia.ulb.ac.be/irace/`.

Para evitar el sobre-entrenamiento (*overfitting*), el ajuste fue hecho automáticamente con datos de otras ciudades: Clorinda, Puerto Iguazú y Pampa del Indio.

Support Vector Regressor (SVR)

Las *Support Vector Machines* son una clase de técnica supervisada que construyen tanto reglas de decisiones lineales como no-lineales y modelos de regresión. En este caso se utilizó el algoritmo `SVR` del módulo `SVM`. Éste método implementa una regresión *Epsilon-Support Vector*. Luego de la etapa de ajuste de hiperparámetros, el valor para la penalidad es de $C = 0{,}887453$, y el núcleo de función de base radial (RBF) con un valor de $gamma = 0{,}015561$.

Perceptron Multicapa (MLP)

Las redes neuronales son contruidas a partir de una gran cantidad de unidades sencillas altamente conectadas entre sí. Ellas pueden ser entrenadas para generar aproximadores universales de funciones. Se utilizó la clase `MLPRegressor` del módulo `neural_network`. Ésta clase implementa una técnica de regresión utilizando un MLP. Para ello optimiza el error cuadrático utilizando el *LBFGS* y el descenso estocástico por grandiente. Luego de la etapa de ajuste, se obtuvo un valor para el término de regularización cuadrática de $alpha = 0{,}070921$, y un total de tres capas y con tres neuronas cada unas completan la arquitectura del modelo. La activación es hecha por la función lineal rectificada $f(x) = max\{0, x\}$.

Regresión de K-Vecinos Más Cercanos (KNNR)

Se utilizó la clase `K-NeighborsRegressor`. Este método infiere una regresión basada en los k-vecinos más cercanos. El objetivo es predicho por una interpolación local de los objetivos en el entorno de vecinos del conjunto de datos de entrenamiento. El conjunto de datos original se descompuso utilizando componentes principales (*PCAs* [41]), sólo cinco fueron usados. Luego de la etapa de ajuste, se obtuvo que el número de vecinos $n\_neighbors = 4$, la función de pesos utilizada en predicción que resultó generar mejor desempeño fue $weights = uniform$, la métrica utilizada para medir la distancia fue $metric = Chebyshev$ y el algoritmo que nuestro modelo utiliza para calcular los vecinos más cercanos resultó ser $algorithm =$ "*brute*".

Regresión de Árboles de Decisión (DTR)

Los árboles de decisión son reglas de clasificación construidas de forma incremental, a partir de las cuales se puede aprender un modelo de regresión. En este trabajo se utilizó la clase `K-NeighborsRegressor` del módulo `tree`. Nuevamente, se utilizó *PCA* pero, esta vez, se conservaron sólo los dos primeros componentes. Además, luego de dicha etapa se concluye que la regla de división sea $splitter =$ "*best*", el valor máximo para la profundidad del árbol $max\_depth = 3$ y el mínimo valor de muestras requeridas para dividir un nodo interno es $min\_samples\_leaf = 5$.

La elección del número de componentes en el *PCA* para los dos últimos métodos fue basada en prueba y error, buscando por el subconjunto más pequeño que produzca buenos

resultados.

3.3. Evaluación y análisis de los modelos generados

Cabe aclarar que en todas las figuras siguientes, las últimas 40 semanas no fueron utilizadas para construir los modelos, por lo que han sido completamente predichas.

La Figura **3-6** muestra los resultados tanto del modelo lineal clásico como el *Ridge*. Estos resultados concuerdan con estudios previos. Ambos regresores lineales producen resultados muy similares, por lo que resulta evidente que es preferible utilizar el primero debido al menor costo computacional que requiere.

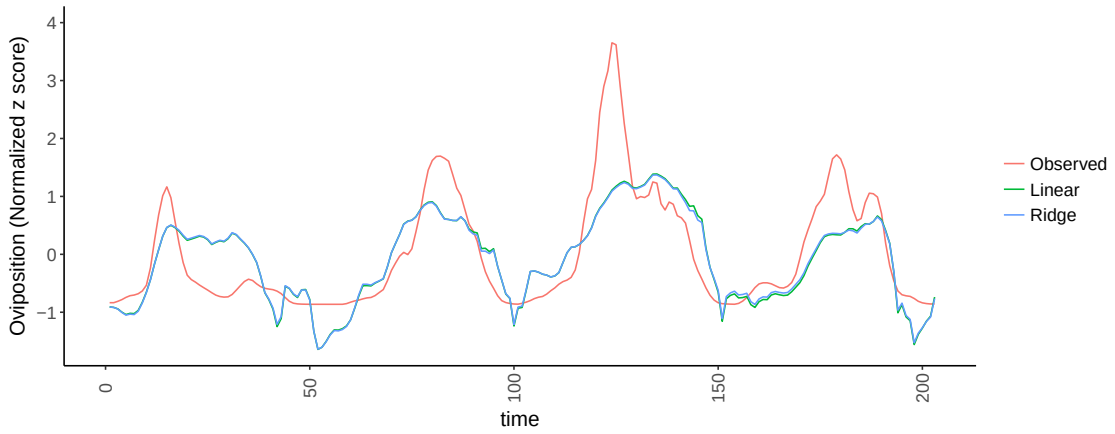


Figura 3-6.: *Z-score* observado, regresiones lineales tradicional y *Ridge*.

Los regresores lineales no se adecúan a los picos de los datos observados, y tienden a subestimar los valor más pequeños.

La Figura **3-7** muestra los datos observados y el resultado del procedimiento de regresión por *SVR*. Este último no modela los picos de los primeros, pero si produce un ajuste relativamente bueno en la mayor parte de los datos.

La Figura **3-8** muestra los resultados del ajuste de los datos observados utilizando la técnica de *MLP*. Dicho ajuste es muy bueno, aunque el modelo sobreestima los datos alrededor de la semana 25 del estudio, y alrededor del último pico.

La Figura **3-9** muestra los resultados producidos por el procedimiento de *KNN*. Se observa que, también, este modelo es muy bueno aunque falla siguiendo a los dos picos más altos. El primero, alrededor de la semana 125 es subestimado, y el segundo, que está cerca de la semana 180, es sobreestimado.

La Figura **3-10** muestra el resultado de aplicar *DTR*. La estructura de este procedimiento produce salidas planas que, sin embargo, siguen de cerca los datos observados.

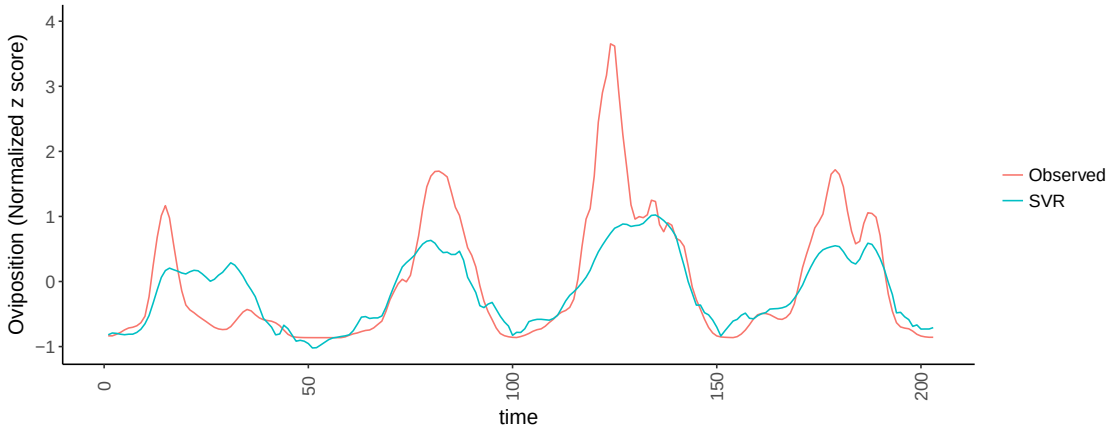


Figura 3-7.: *Z-score* observado y regresión SVR.

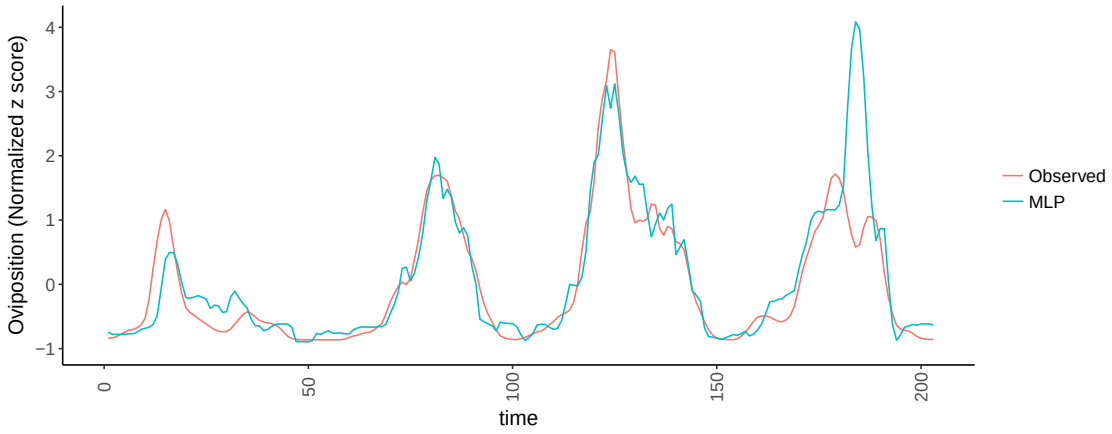


Figura 3-8.: *Z-score* observado y regresión MLP.

La tabla **3-1** presenta un resumen de los datos observados y los ajustados: los valores mínimos (Mín) y los máximos (Máx), el primer ($q_{1/4}$) y tercer cuartil ($q_{3/4}$), la mediana ($q_{2/4}$) y la media.

La Tabla **3-1** revela los siguientes hechos:

- Las regresiones lineal y *Ridge* exageran los mínimos, ya que producen valores que son aproximadamente el doble que los observados.
- El Perceptron Multicapa exagera el máximo por alrededor de un 10 %, mientras que los otros modelos subestiman dicho valor. Notar que el *SVR* aplana el máximo por un factor de aproximadamente 3.6.
- La media y mediana observadas difieren notablemente, sugiriendo que los modelos están significativamente sesgados a la izquierda.

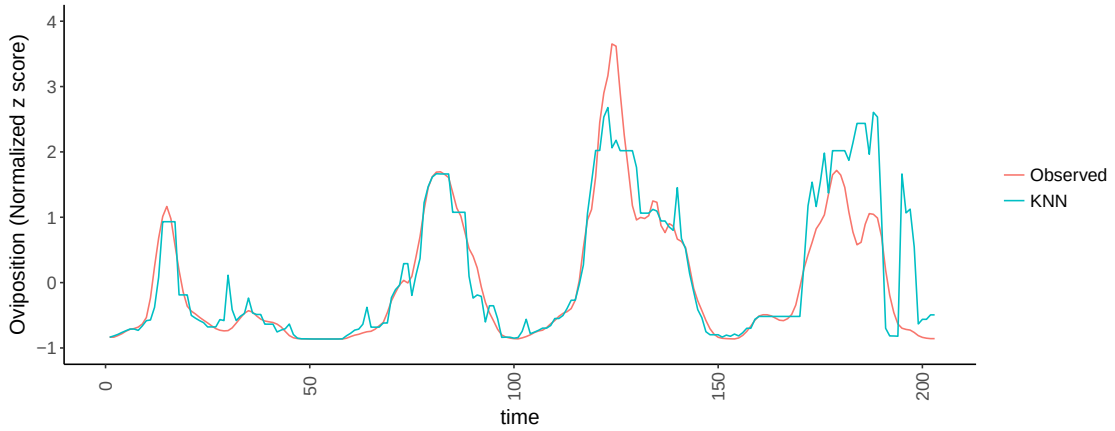


Figura 3-9.: *Z-score* observado y regresión KNN.

Tabla 3-1.: Resumen de los datos observados y los ajustados

	Mín	$q_{1/4}$	$q_{1/2}$	Media	$q_{3/4}$	Máx
Observado	−0,863	−0,742	−0,487	0,000	0,704	3,652
Lineal	−1,641	−0,716	0,027	−0,087	0,462	1,387
Ridge	−1,638	−0,680	0,028	−0,084	0,459	1,370
MLP	−0,894	−0,677	−0,323	0,093	0,716	4,084
DTR	−0,752	−0,752	−0,128	0,138	0,998	2,312
KNNR	−0,863	−0,699	−0,501	0,099	1,033	2,679
SVR	−1,021	−0,601	−0,232	−0,147	0,309	1,023

- El valor más cercano al observado, de la mediana, es producido por *KNN*, el cual también conduce a un valor cercado de la media.

La Figura **3-11** muestra los datos observados y predichos como un *scatterplot*. Esta figura revela que ninguno de los modelos es capaz de seguir los valores más grandes observados, y que los modelos Lineal, *Ridge* y *SVR* son los menos aptos para esta tarea, mientras que *MLP* es la más adecuada. Además se notó que este último modelo es el más propenso a sobreestimar los datos. Cabe destacar que la subestimación es, para el punto de vista de la aplicación, más peligroso que la sobreestimación, dado que el primero tiende a ser un falso indicador negativo que puede llevar a no disparar medidas de prevención en casos en que efectivamente se necesiten.

A continuación se analizarán los errores. Las Figuras 3.12(a) y 3.12(b) muestran, respectivamente, los histogramas y boxplots de los errores producidos por cada modelo. Los errores generados por *KNN* son los más concentrados alrededor de cero, seguidos por el *MLP*. Los dos errores más extendidos son se corresponden con las regresiones lineales. Éste es un indi-

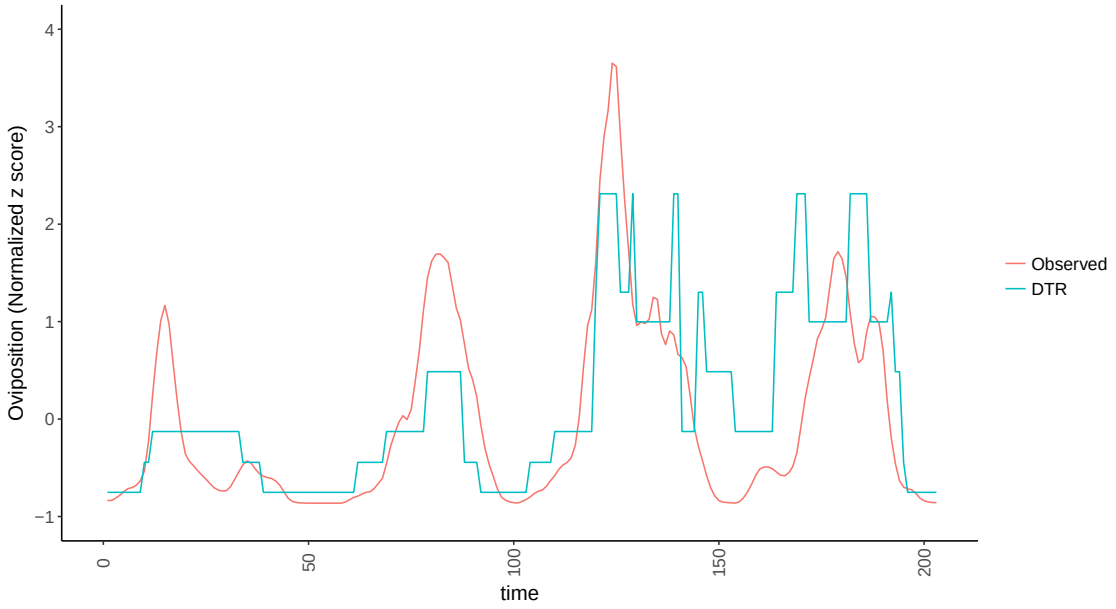


Figura 3-10.: *Z-score* observado y regresión DTR.

cador de que los modelos obtenidos utilizando simples técnicas lineales son los peores entre los considerados en este trabajo.

La Tabla **3-2** presenta las medidas de calidad de los modelos aquí considerados: los coeficientes de la correlación de *Pearson* [87] entre los valores observados y ajustados, usando el conjunto de datos completo (Corr11) y sobre el 20 % para la validación (CorrL20); además, el Error Cuadrático Medio sobre el conjunto de datos completo (MSE), y sólo sobre los datos de validación (MSEL20). Siguiendo [19], también se incluyó el *z-score* medio obtenido de la validación cruzada y su desviación estándar (SD del Z-Score).

Tabla 3-2.: Medidas de calidad de los modelos

	Corr11	MSE	Z-Score Medio	SD del Z-Score	CorrL20	MSEL20
Lineal	0,774	0,624	1,108	0,278	0,890	0,580
Ridge	0,775	0,621	1,072	0,277	0,896	0,566
SVR	0,837	0,613	**0,834**	0,490	**0,967**	**0,464**
MLP	0,875	0,528	1,086	0,288	0,727	1,023
KNN	**0,888**	**0,494**	0,981	0,362	0,797	0,936
DTR	0,679	0,768	1,148	0,544	0,532	1,131

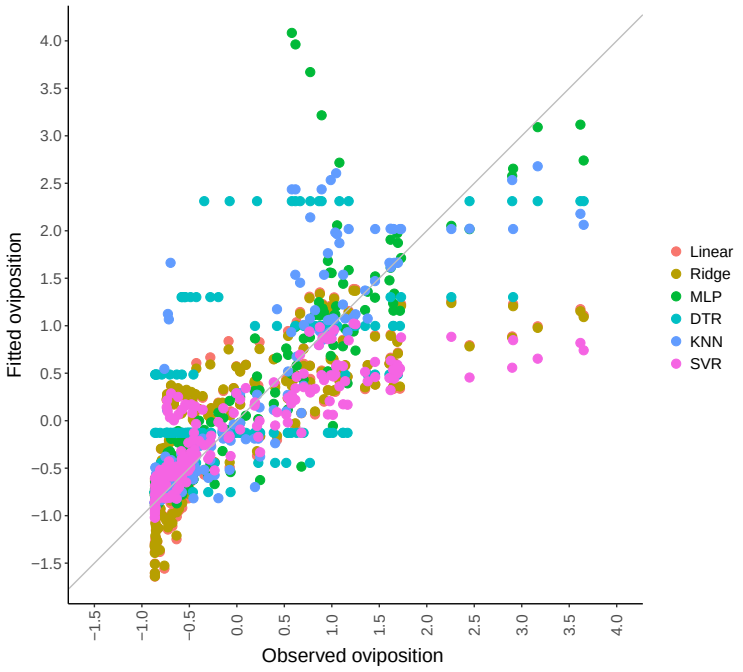


Figura 3-11.: Scatterplot de los valores observados y predichos.

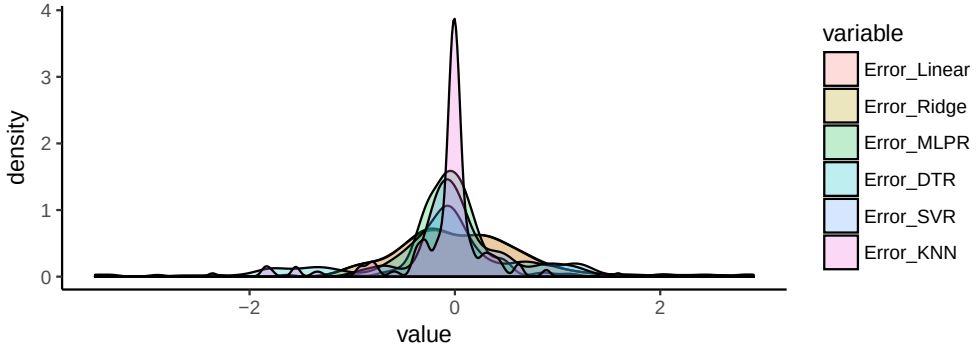
3.4. Discusión de resultados obtenidos en la primer etapa

Un punto interesante que aparece en los resultados es que todos los modelos aquí presentados ajustan bien en los patrones principales pero no necesariamente en los picos extremos. Una hipótesis es que la población del vector se desconecta de las variables macroambientales/climáticas cuando las condiciones son óptimas y, nuevamente, se restringe cuando las condiciones ambientales son subóptimas. De hecho, es razonable el hecho de que no podemos esperar ajustar exactamente la población urbana del vector sólo basándonos en variables macroambientales a gran escala.

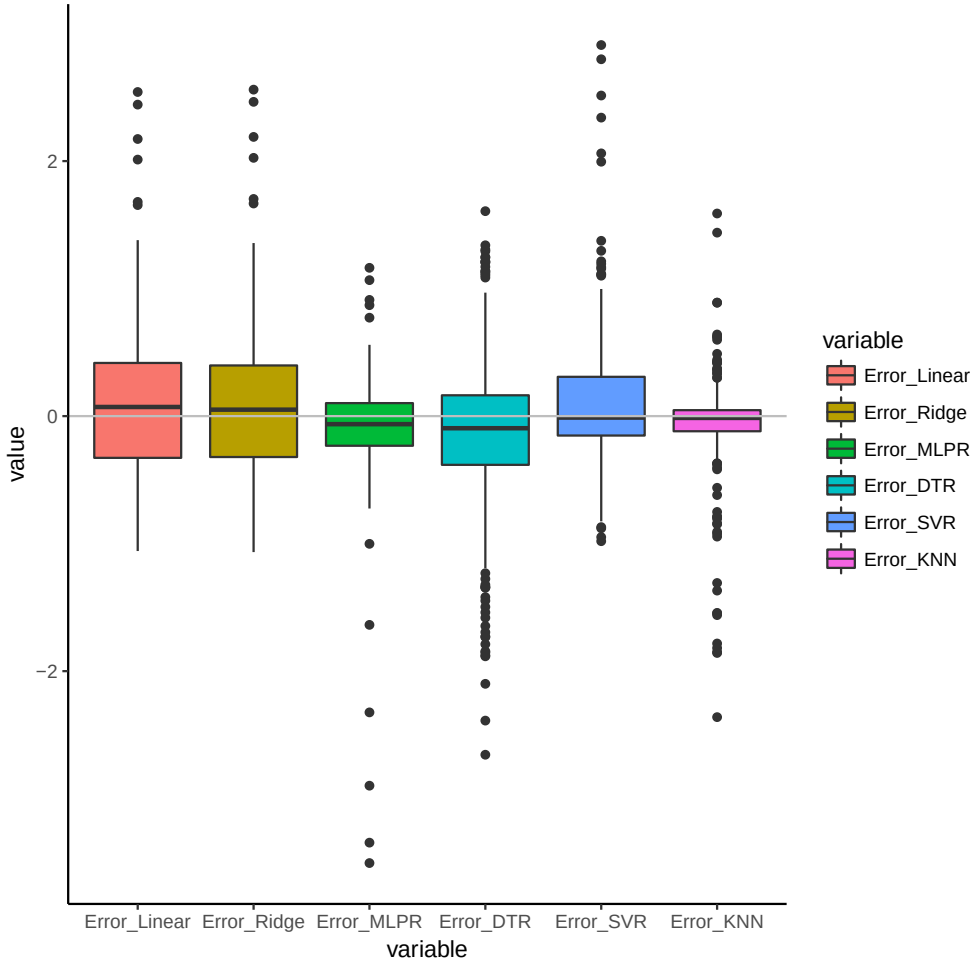
Teniendo en cuenta la bondad de ajuste incluida en las Tablas **3-1** y **3-2**, y un análisis de errores, podríamos considerar que *KNN* aparenta ser el mejor método para este problema. Tiene una correlación cercana al 90 %, considerablemente mayor al 75 %, el valor típico obtenido por las técnicas lineales.

El valor medio del *z-score* llevaría a elegir el *SVR* como la mejor técnica [14]. Cabe destacar que la desviación estándar de esta medición de calidad es tan alta que es poco probable que sea una buena elección en sí misma. Por esta razón, seguimos un enfoque holístico en las próximas conclusiones.

Si tenemos en cuenta las seis métricas de la Tabla **3-2**, es clara la conclusión de que los mejores métodos para modelar la población del vector basada en variables ambientales deri-



(a) Histogramas



(b) Boxplots

Figura 3-12.: Errores

vadas de información satelital son: *K-vecinos más cercanos*, *Perceptron Multicapa* y *Support Vector Machine*.

3.5. Problemáticas de un sistema regional de modelado de poblaciones de mosquito

La tarea de modelar la población de mosquitos a nivel regional trae consigo numerosas problemáticas, algunas de las cuales ya fueron mencionadas en la sección de Motivación y Marco Teórico.

En este trabajo, un problema que encontramos es la escasez de datos de campo que existen: el desempeño de estos algoritmos se podría mejorar sustancialmente utilizando conjuntos de datos más grandes. Aunque el período utilizado es grande en comparación con trabajos similares sobre la población vectorial, dicho conjunto sigue siendo muy pequeño desde el punto de vista del aprendizaje automático.

Otro problema, no menos importante teniendo en cuenta el objetivo final de los esfuerzos puestos en este sentido (tener modelos operativos de riesgo), es la gran escasez de puntos (ciudades) de los cuales se posee información de campo (oviposición). Esto resulta un problema dado que si los modelos son entrenados con datos de un punto geográfico **A** (Tartagal, por ejemplo), a priori no podemos asegurar que serán capaces de ajustar correctamente al comportamiento de la variable objetivo de un punto **B** (Córdoba, por ejemplo). Ni siquiera se poseen datos para validar dicha conducta. Esto limita el alcance regional de las herramientas de este tipo.

4. Generalización espacial de modelos epidemiológicos basada en el concepto de Distancia Ambiental Normalizada NED

Los modelos temporales descriptos en capítulos anteriores se basan en la generación de relaciones empíricas entre datos ambientales derivados de información satelital y los datos de campo, correspondientes a los del vector propiamente dicho. Esto significa que sólo pueden construirse modelos en lugares donde esté disponible la información de campo, problema que se menciona al concluir el capítulo anterior.

En ese marco, y con el objetivo final de mejorar la aplicación operativa presentada por Porcasi y colaboradores en 2012 [71], en este capítulo se plantea el objetivo específico de generar una metodología para espacializar los datos contruidos siguiendo la metodología del capítulo anterior, basada en el concepto de ***Distancia Ambiental Normalizada*** (NED).

4.1. Descripción del problema

A partir de la disponibilidad de datos de campo en N localidades diferentes, se generan N modelos que relacionan la oviposición con variables ambientales derivadas de datos satelitales (*lst_night*, *lst_day*, *ndvi*, *ndwi*, *prec*). Por simplicidad, sin pérdida de generalidad, supongamos que dichos modelos son lineales:

$$ovip_j = \beta_j + \sum coef_{ji} \times envVar_i(j) \tag{4-1}$$

donde $coef_{ji}$ representa los coeficientes del modelo de la ciudad j para la variable i, y $envVar_i(j)$ representa la variable ambiental i evaluada en la posición correspondiente a la ciudad j. Es decir que para cada ciudad j, hay un conjunto diferente de coeficientes, que son aquellos que generan un ajuste óptimo de los datos disponibles. Aquí se denominan a estos N modelos: $M_1,\ M_2,\ \ldots,\ M_N$.

Así, el problema que se plantea es aquel en el que el modelo se debiera utilizar en una nueva ciudad (no incluida en las N anteriores) en donde se quiere obtener una estimación de la abundancia del vector; para de esa manera, obtener la estimación mencionada para

cualquier otra ciudad. En particular, en este caso, en la región norte de Argentina donde no se disponga de datos de campo.

La idea más simple para extrapolar los modelos obtenidos sería usar, para un punto/-pueblo adicional localizado en la posición X, un modelo M_X igual al modelo conocido de la ciudad más cercana geográficamente (vecino más cercano) es decir $M_X = M_J$ donde J corresponde a la ciudad más cercana. Una mejora a este enfoque, es utilizar un promedio de los N modelos conocidos ponderados por el inverso de la distancia de este nuevo punto X a cada una de las ciudades J donde se dispone de un modelo. Es decir, el modelo de la ciudad más cercana pesará más y el de la más alejada pesará menos, es decir:

$$M_X = \sum \frac{M_j}{L_j} \tag{4-2}$$

donde L_j representa la distancia normalizada de la ciudad J a X (en términos de la localización geográfica de la nueva ciudad).

El problema de las soluciones anteriores, es que en realidad es más razonable pensar que el comportamiento de la población del vector/mosquito en una ciudad en el punto X será más coincidente con una que se encuentre en una ciudad que sea más similar **ambientalmente** y no necesariamente con aquella que está más cerca geográficamente. En ese sentido, se debería utilizar (en el esquema de vecino más cercano) el modelo de la ciudad J que posea el medio ambiente más similar al del punto X. En otras palabras, así como "más cerca", significa típicamente coordenadas geográficas (o posiciones) similares; en el sentido ecológico/ambiental, podemos pensar "más cercaçomo que sus variables ambientales son similares. De esta forma aparece naturalmente el concepto de ***Distancia Ambiental***.

4.2. Distancia Ambiental Normalizada (NED)

El concepto de **Distancia Ambiental**, si bien no es completamente nuevo, no ha sido utilizado en el contexto de la epidemiología. Una revisión exhaustiva de bases de datos bibliográficas de revistas indexadas nos arroja que sólo existen 11 publicaciones con "*Environmental Distance*" en su título. El más citado de éstos, es el trabajo de Hirzel [39] quien utiliza esta idea en el contexto del estudio de ecología y distribución de especies.

Con un enfoque similar, podemos encontrar las contribuciones de Krasnova, Mendez y Faber [24, 45, 53]. En estos trabajos el concepto de nicho ecológico está ligado naturalmente a la idea de compartir condiciones ambientales que hacen de un lugar determinado un sitio apto para que una determinada especie pueda desarrollarse. Una acepción completamente diferente de "Distancia Ambiental" puede encontrarse por ejemplo en [58], donde ésta se relaciona a la percepción cognitiva del ser humano con su entorno.

Si se relaja la búsqueda a la aparición de "Environmental Distance" en el título, palabras claves o resumen de los trabajos, se pueden encontrar 164 contribuciones que pertenecen pri-

mordialmente a las áreas de ciencias de la tierra, genética, agricultura y ciencias biológicas. Sólo 10 están declaradas como ligadas a la medicina, pero ésto es a través de estudios genéticos. Aquí podemos encontrar tan sólo un par de contribuciones [1, 91] que indirectamente relacionan, a través de las ideas de la eco-epidemiología y la distribución de especies vectores de malaria, las ideas de "Distancia Ambiental" con la problemática epidemiológica.

4.2.1. Solución propuesta

Para poder aplicar las ideas ya discutidas, que biológicamente aparecen como razonables, es necesario definir las variables involucradas en el concepto "similitud ambiental" y luego definir una **distancia** ambiental. Para el primer caso, se utilizaron las 19 variables bioclimáticas incluidas en *WorldClim* [38], construidas a partir de una gran serie de tiempo (1950−2000). Además se incluyeron valores medios mensuales de *NDVI* de *MODIS* (durante un período de 10 años, 2005 − 2014).

Una vez seleccionadas las variables ambientales, basado en ellas, se define la distancia generalizada $dist_{x_1 \ - \ x_2}$ entre dos posiciones geográficas arbitrarias x_1 y x_2 como:

$$dist_{x_1 \ - \ x_2} \ = \ \sqrt{\sum (v_{k_1} - v_{k_2})^2} \tag{4-3}$$

donde v_k son las 19 variables bioclimáticas más altitud y los *NDVI* mensuales medios.

De esta manera, finalmente se puede estimar la distancia ambiental de cada ciudad en una ubicación X, y las 4 ciudades modelables, Js, y volver a calcular el método de extrapolación de la ecuación 4-2 pero ahora usando la distancia ambiental. Aquí se ha tomado $N \ = \ 4$ ya que en la realidad se cuenta sólo con 4 ciudades con series completas de datos para modelar: Pampa del Indio, Clorinda, Tartagal y Puerto Iguazú (Fundación Mundo Sano[1].

Operativamente, para calcular las distancias, definimos una región de 20 km alrededor de cada ciudad J para caracterizar las variables de estas ciudades (como una media de los píxeles en este buffer). Luego, utilizamos la probabilidad de pertenencia (clasificación supervisada) a cada clase utilizando software *ENVI* para calcular la NED de cada píxel a cada una de las 4 ciudades modeladas.

Cabe mencionar aquí que por *normalizada* entendemos que la suma de las inversas (es decir los pesos con los que cada modelo individual interviene) es igual a 1:

$$1 \ = \ \sum \frac{1}{L_j} \tag{4-4}$$

Como ejemplo de las variables ambientales utilizadas en el cálculo de la NED algunas de ellas son presentadas en las Figuras **4-1** y **4-2**. La Figura **4-1** muestra en RGB la temperatura media anual, el rango de temperatura y la precipitación anual. Aquí claramente

[1] `https://www.mundosano.org/`

puede apreciarse, tanto la zonificación de la región de estudio marcando áreas ambientalmente similares y diferentes, como la baja resolución espacial de los productos *WordClime* utilizados.

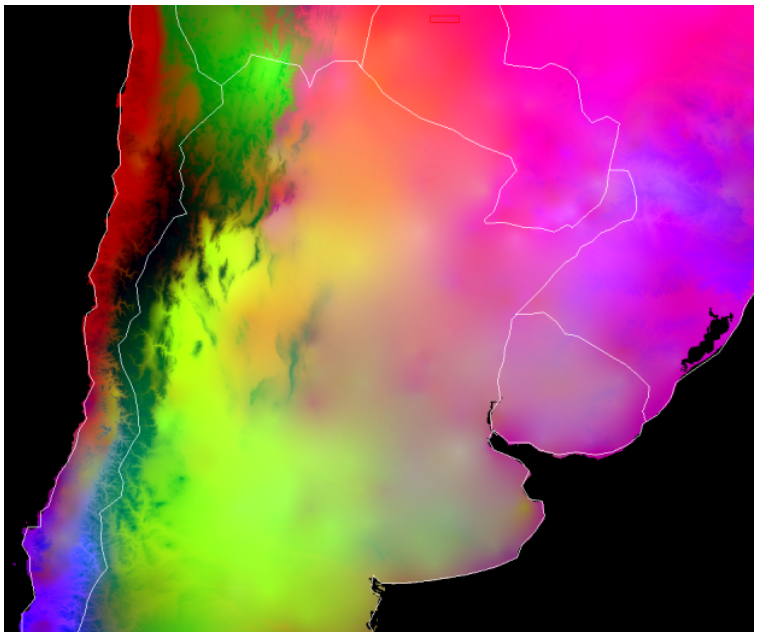


Figura 4-1.: RGB: BIO1 = temperatura media anual, BIO7 = rango anual de temperatura (BIO5-BIO6) y BIO12 = precipitación anual, correspondientemente.

De una manera similar, la Figura **4-2** presenta en RGB el *NDVI* de *MODIS* promedio de Enero, *NDVI* de *MODIS* promedio de Julio y el *DEM*.

4.3. Evaluación de la solución propuesta

El resultado de las distancias ambientales normalizadas calculadas de cada pixel a cada una de las 4 ciudades se presenta en la Figuras **4-3**, **4-4**, **4-5** y **4-6**. Es importante tener en cuenta que la inversa de Distancia Ambiental Normalizada ($\frac{1}{NED}$) nos dice qué tan similar es una ciudad en comparación con las otras tres.

Claramente por estar normalizada, no es una medida de similaridad en términos absolutos. Así, por ejemplo en la Figura **4-4** los píxeles con valores cercanos a 1 significan que estos lugares son ambientalmente mucho más parecidos a Iguazú que a Tartagal o Clorinda o Pampa del Indio.

Sólo como ejemplo, la inversa de la distancia normalizada ($\frac{1}{NED}$) de Tucumán, Corrientes y Salta se describen en la Tabla **4-1**. Estos valores son presentados de una manera diferente en la Figura **4-7** donde se intenta graficar con más claridad la contribución que tendrán cada uno de los 4 modelos previamente desarrollados, cuando se intenten modelar estas 3 nuevas ciudades.

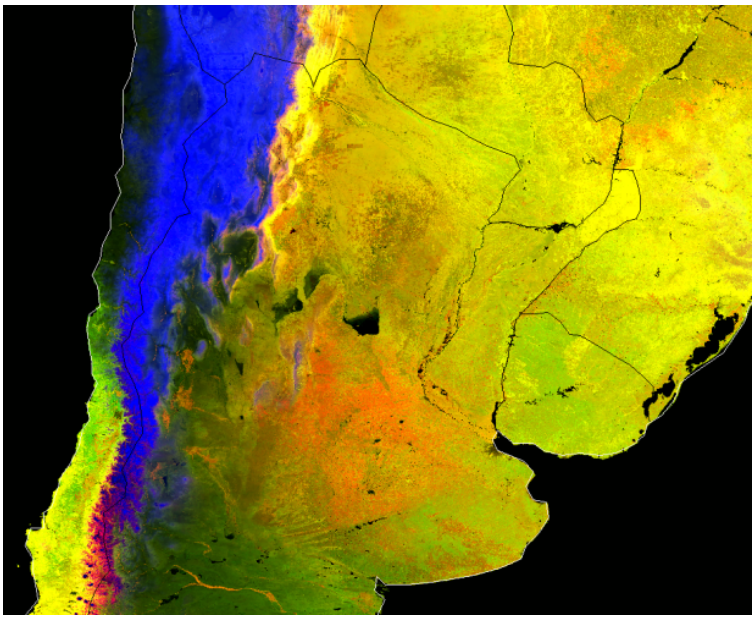


Figura 4-2.: RGB: *NDVI* de *MODIS* promedio Enero, *NDVI* de *MODIS* promedio de Julio y DEM, correspondientemente.

Para el caso de las tres ciudades simuladas, la distancia ambiental realmente tiene una fuerte correlación con la distancia geográfica estándar. Estos ejemplos muestran más bien cómo el método funciona, que las ventajas que ofrece el mismo. Sin embargo la diferencia entre la distancia ambiental y la geográfica puede observarse claramente en las Figuras **4-3** a **4-6**. En todas ellas vemos cómo la distancia ambiental posee bordes abruptos (cosa que la distancia geográfica nunca tendrá). Por ejemplo en la Mesopotamia (Figura **4-4**) o en las yungas (Figura **4-3**) puede verse cómo pequeñas distancias geográficas tienen asociadas fuertes gradientes en la distancia ambiental (grandes distancias ambientales) y por ende podríamos suponer que localidades cercanas geográficamente poseen patrones temporales de la población de vectores muy diferentes.

Tabla 4-1.: Inversa de NED de cuatro ciudades a las 4 localidades predefinidas

	Puerto Iguazú	Clorinda	Pampa del Indio	Tartagal
Tucuman	0,197	0,011	0,388	0,402
Corrientes	0,491	0,466	0,039	0,002
Salta	0,112	0,005	0,133	0,749

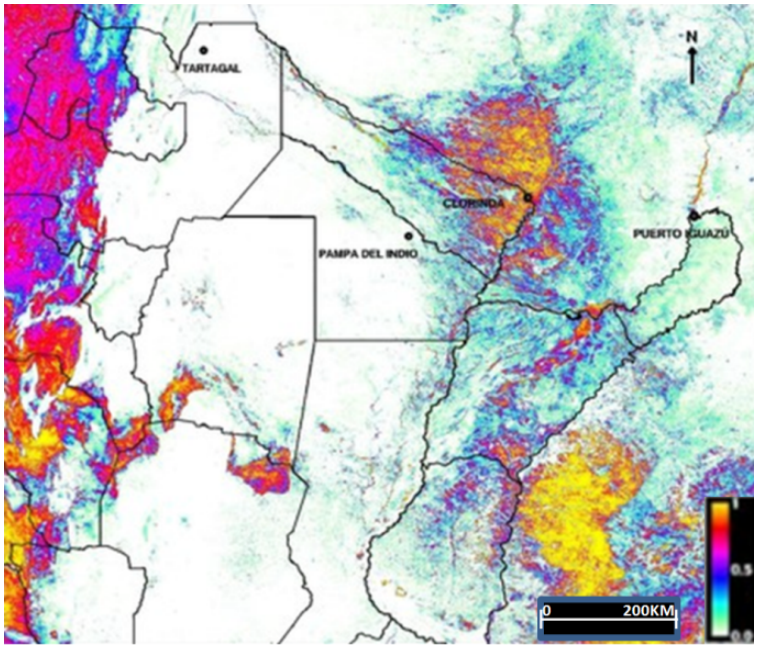


Figura 4-3.: Similaridad ambiental ($\frac{1}{NED}$) de cada pixel a las condiciones de Clorinda.

4.4. Discusión y propuesta futura

En el presente capítulo pretendemos generar una contribución al objetivo de construir pronósticos dinámicos para la población de vectores, utilizando productos satelitales. Se aborda el problema de cómo generalizar espacialmente modelos ajustados para ciertas localidades específicas.

Para ello, se propone una metodología basada en ideas ecológicas incorporando el concepto de distancia ambiental normalizada. Se ha mostrado que el mismo si bien es novedoso es conceptualmente simple. Se describe un método simple para calcularlo y ejemplos de su implementación especifica de la estimación de este parámetro en función de un conjunto de variables ambientales relevantes para la ecología del vector del dengue.

Cabe resaltar que esta idea de interpolación ambiental que se plantea aquí para modelos relacionados a la epidemiologia, podría también ser utilizados a la hora de contar con datos puntuales de otras variables (por ejemplo rendimiento en la producción agrícola) donde la distancia espacial no refleje tan adecuadamente la similaridad entre sitios como lo es la distancia ambiental.

Los resultados y metodologías aquí planteadas fueron presentadas en el *Congreso Bienal de IEEE Argentina* (ARGENCON) durante la primer semana de Junio de 2018. La presentación fue realizada bajo el título *Generalización espacial de modelos epidemiológicos basada en el concepto de Distancia Ambiental Normalizada NED* [84] y se podrá encontrar en la *IEEE Xplore Digital Library*[2].

[2] `https://ieeexplore.ieee.org/`

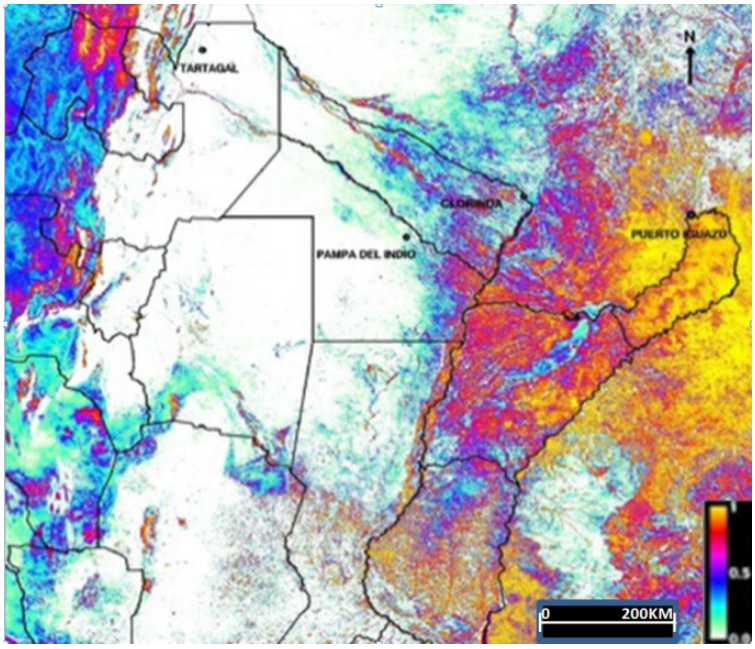


Figura 4-4.: Similaridad ambiental ($\frac{1}{NED}$) de cada pixel a las condiciones de Iguazú.

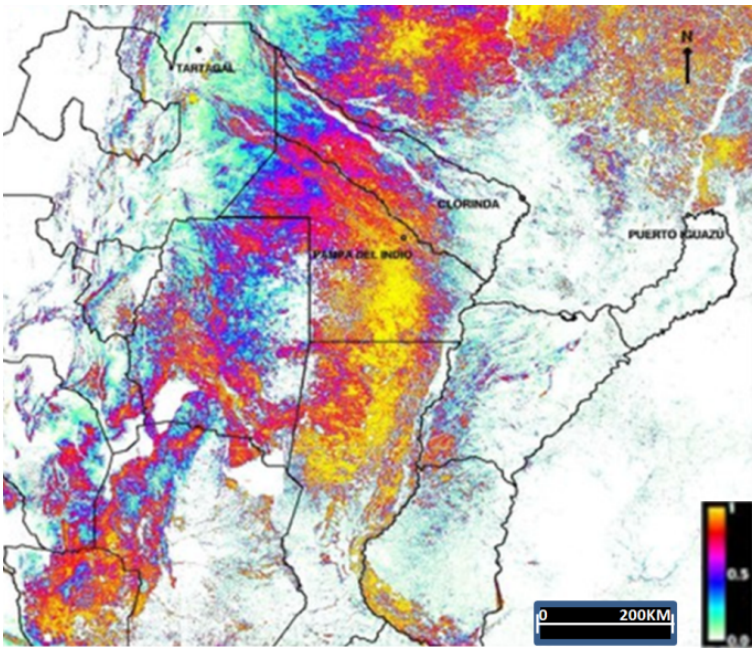


Figura 4-5.: Similaridad ambiental ($\frac{1}{NED}$) de cada pixel a las condiciones de Pampa del Indio.

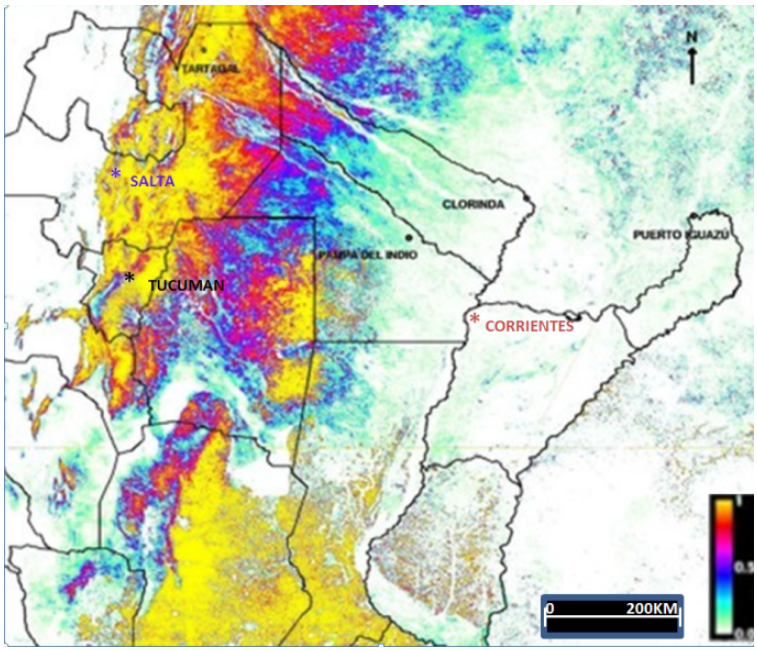
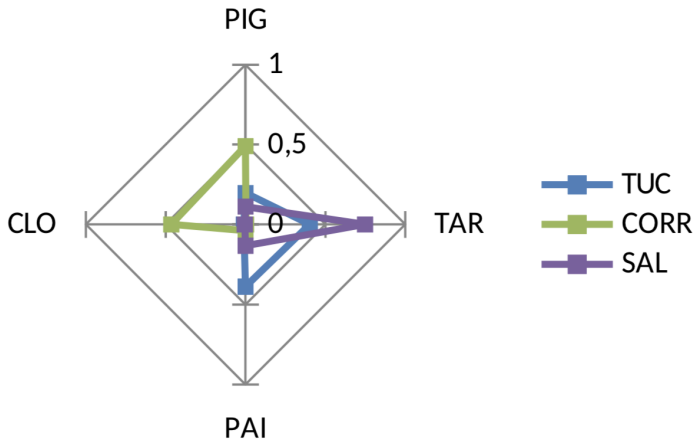


Figura 4-6.: Similaridad ambiental ($\frac{1}{NED}$) de cada pixel a las condiciones de Tartagal. Aquí también se incluyen las localizaciones de las tres ciudades tomadas como ejemplo para el cálculo de nuevos modelos (Salta, Tucumán, Corrientes).



[H]

Figura 4-7.: Contribución que poseen los modelos para Salta, Corrientes y Tucumán de los 4 modelos disponibles.

5. Discusión y Conclusiones

Dengue, Chikungunya y Zika son enfermedades virales para las cuales no existen, al día de hoy, vacunas de prevención. Por lo tanto, el control más efectivo proviene de prevenir la propagación del mosquito *Aedes Aegipty* (*Linneaus*). Lo que lleva a la necesidad de saber sobre la dinámica de su población es de suma importancia.

Este trabajo, por un lado, presenta un *framework* de simple utilización para el pronóstico de la oviposición utilizando únicamente variables ambientales extraídas de información satelital y herramientas de Aprendizaje Automático de libre acceso. Y por el otro, establece un concepto novedoso en el área de la epidemiología panorámica cuyo fin es lograr utilizar información de modelado de ciertos puntos geográficos para estimar la abundancia en muchos otros para los cuales no se posee información de campo. Este concepto es el de *Distancia Ambiental Normalizada*.

Las herramientas implementadas en el *framework* son una mejora al sistema operacional de riesgo de Argentina [71]. A su vez, por la arquitectura del mismo, es posible agregar modelos nuevos y modificar las variables independientes a utilizar como predictores (*features*) de una manera sencilla.

En este caso, se utilizaron variables ambientales derivadas de información satelital (temperatura, humedad y precipitación) operacionalmente disponibles para construir modelos temporales capaces de predecir la actividad de oviposición fuera de las casas. En ese sentido, la perspectiva planteada, completamente operativa, implica generar un procedimiento para estimar la actividad del vector y eventualmente independizarse de las mediciones de campo. Dicha contribución se considera de alto valor, entre otras cosas, porque realizar la medición de oviposición en 50 casas todas las semanas, durante largos períodos de tiempo (como se utilizó para generar los modelos) tiene un costo extremádamente alto.

Este estudio resulta ser un avance sobre trabajos previos en el área de la epidemiología panorámica, donde se consideran modelos estadísticos utilizando relaciones lineales [18,20,22] en términos de la capacidad predictiva de los modelos desarrollados aquí. Estas mejoras fueron obtenidas utilizando herramientas de aprendizaje automático que, en este caso, no requieren de un esfuerzo adicional de parte del usuario.

La metodología implementada muestra que algunas herramientas *off-the-shelf* son capaces de manejar las complejas relaciones entre variables, proporcionando así una forma de abordar el importante problema planteado. El mencionado enfoque interdisciplinario proporciona nuevas herramientas para los profesionales que se encuentran trabajando en esta área.

A su vez, este trabajo es un ejemplo de cómo el uso de herramientas automáticas para la configuración de algoritmos, como *iRace* pueden reducir la complejidad del ajuste de hiperparámetros de los modelos y proveer un marco de referencia para la selección de los mismos. Adicionalmente, se muestra la importancia de la utilización de la Validación Cruzada (VC), raramente utilizada en usuarios del Sensado Remoto. Utilizamos VC para disminuir la dependencia de los resultados de evaluación sobre una selección particular de los conjuntos de entrenamiento y validación en la etapa de elección del modelo. Aqui se utiliza un procedimiento particular de VC para series de tiempo. Todos los modelos aquí discutidos pueden ser ejecutados con *scripts* de Python disponibles libremente[1].

En lo que respecta a la comparacion de algoritmos, se encontró que la Regresión por K-Vecinos Cercanos (KNNR), el Perceptron Multicapa (MLP) y la *Support Vector Machine* resultan ser los modelos predictivos de la población de vectores que mejores resultados arrojan.

A pesar de que el período utilizado es largo en comparación con trabajos similares sobre poblaciones del vector, el desempeño de estos algoritmos puede ser mejorado sustancialmente utilizando conjuntos de datos más grandes. Otra manera de mejorarlos seria realizar ajustes más finos en el modelado y/o bien utilizar otras técnicas de mayor complejidad dentro del área del Aprendizaje Automático.

La otra contribución importante de este trabajo está relacionada con la necesidad de poseer modelos de oviposición para distintas ciudades, evitando el gran costo de la recolección de datos y el entrenamiento para cada una de las ciudades o puntos para los cuales se quiera poseer datos. Aquí presentamos una forma de establecer relaciones entre los distintos lugares geográficos teniendo en cuenta las características ambientales que poseen. La hipótesis más fuerte que asumimos es la que nos dice que el comportamiento de los vectores está altamente correlacionado (al menos dentro de cierto rango) a las características ambientales del punto en el que que se observa.

Así se presenta, desarrolla e implementa el concepto de Distancia Ambiental Normalizada, el cual permite llevar a cabo lo mencionado en el párrafo anterior estableciendo una distancia vectorial utilizando el espacio de características ambientales extraídas de información satelital, en vez del espacio geográfico.

En conjunto, ambas contribuciones aportan un muy alto valor de capacidad de mejora al sistema operacional de riesgo de la república Argentina. A su vez, en perspectiva, aporta valor a la proyección de mejora de dichos modelos por su facilidad de uso y extrapolación a distintas zonas.

Otro punto de valor del trabajo es el caracter integrador e interdisciplinario del mismo, demostrando la utilidad y la necesidad de la inserción del Aprendizaje Automático en áreas de impacto social.

A su vez, cabe destacar que lo desarrollado involucra conocimientos de diversas áreas de las Ciencias de la Computación abarcando temáticas, por ejemplo, de Ingeniería del Software,

[1] `https://github.com/juansca/modeling-mosquitos`

a la hora de realizar el análisis de requerimientos, generación de la arquitectura y establecer la metodología de trabajo. Por otra parte, también se utilizan conocimientos de estadística, modelos y simulación y distintas áreas de matemática para el entendimiento de los distintos algoritmos, tomar decisiones con respecto a ellos y a las hipótesis y conclusiones. Muchos otros conceptos aprendidos a nivel general por las distintas materias han sido aplicados en el desarrollo. Es por esto que me resulta de suma importancia mencionar que lo realizado en este trabajo, con las características interdisciplinarias y la envergadura del mismo, me permitió integrar, de una manera muy contructiva para mi desarrollo profesional, todo lo aprendido y adquirido a lo largo de la carrera.

Es importante resaltar finalmente que los resultados y metodologías incluídos en este trabajo de grado han dado lugar a tres publicaciones indexadas en la base de datos scopus, a saber:

- J. M. Scavuzzo, F. Trucco, M. Espinosa, C. B. Tauro, M. Abril, C. M. Scavuzzo, and A. C. Frery. Modeling dengue vector population using remotely sensed data and machine learning. Acta tropica, 185:167–175, 2018. [86]
- J. Scavuzzo, M. Espinosa, E. Di Fino, M. Abril, G. Peralta, and C. Scavuzzo. Generalización espacial de modelos epidemiológicos basada en el concepto de distancia ambiental normalizada ned. 2018. [84]
- J. Scavuzzo, F. Trucco, C. Tauro, A. German, M. Espinosa, and M. Abril. Modeling the temporal pattern of dengue, chicungunya and zika vector using satellite data and neural networks. volume 2017-January, pages 1–6, 2017 [85]

A. Anexo: Detalles del código

Se decidió realizar todo el desarrollo en el lenguaje de programación *Python* por su simplicidad, buen desempeño y su extensa comunidad activa. Esto facilita el desarrollo e incrementa la velocidad de producción. El proyecto está disponible en `https://github.com/juansca/modeling-mosquitos` y en su sección inicial se pueden encontrar instrucciones para su instalación. A continuación se describirán algunos aspectos que se consideran relevantes de los distintos módulos, sin entrar en detalles.

El módulo `data` por un lado tiene un archivo llamado `constants.py` en donde se definen algunas constantes que son dependientes del conjunto de datos que se utilizará. Es importante dado que es allí en donde se especifican los *features* (o columnas) que se utilizarán como input para predicción. A su vez, en dicho módulo, el archivo `data_cleaner.py` posee una clase llamada `DataCleaner` que es la encargada de realizar la limpieza de los datos. Para realizar la limpieza de los datos se debe ejecutar el script `scripts/clean_data.py`, el cual arroja el siguiente instructivo:

```
$ python data/scripts/clean_data.py --help

Clean Data.

Usage:
  ./clean_data.py -i <file> -o <dir> [--p_eval <float>] [--instances <n>]
      [--overlap <f>]

Options:
  -i <file>            Evaluate dataset path
  -o <dir>             Directory where the evaluation plot result will be
                       saved
  --p_eval <float>     Percentage to evaluation dataset. [default: 0.2]
  --instances <n>      Number of instances to generate from data
                       [default: 1]
  --overlap <f>        Percentage of overlapping between the instances.
                       [default: 0]
```

Por otro lado, el módulo `models` tiene un archivo llamado `models.py` en el cual se declaran los modelos que se utilizarán para el modelado. Es importante que estos modelos sigan la estructura ahí utilizada para que los demás módulos los puedan utilizar correctamente.

Además, allí se encuentra el *script* de entrenamiento, `scripts/train.py`, que entrena el modelo elegido con el conjunto de datos dado e imprime por linea de comandos un conjunto de estadísticas que resultan de realizar validación cruzada sobre los datos brindados por el usuario para dicha tarea. Ésto resulta útil para tener una noción del desempeño del modelo. Éste script devuelve la siguiente documentación de uso:

```
$ python models/scripts/train.py --help

Train a model

Usage:
  ./train.py -i <file> --model <model> [-p <file>]
  ./train.py -h | --help

Options:
  -i <file>       Train/Val dataset path
  --model <model> Model you want to train, is mandatory that it was on
                  models.py file.
  -p <file>       CSV file where are saved the hyperparameters
                  (in case of tunning module was used).
```

Por otra parte, el módulo posee un *script* de evaluación, que, además de imprimir por linea de comandos el valor del Error Cuadrático Medio de la evaluación, genera un gráfico con la curva real y la curva predicha por el modelo y lo guarda en un directorio. A su vez, guarda un archivo `csv` con los valores reales y los generados por el modelo facilitando así, la posterior manipulación del mismo. La documentación de ayuda para su utilización es:

```
$ python models/scripts/eval.py --help

Evaluate a model

Usage:
  ./eval.py -i <file> -m <model> [-o <file>]

Options:
  -i <file>       Evaluate dataset path
  -m <model>      Model you want to evaluate as pickle format
  -o <dir>        Directory where the evaluation plot result will be saved
```

Finalmente, el sistema desarrollado posee el módulo `tunning`. Allí se realiza el ajuste de hiperparámetros de los modelos. Existe varios archivos en ese módulo que son los que hacen de interfaz con la herramienta *irace*. Algo que cabe destacar aquí es el directorio `parameters`. Allí se colocan los posibles (o intervalos de) valores que generan el espacio

de hiperparámetros donde la herramienta buscará los óptimos para cada modelo. Además, *irace*, usará las instancias de datos en `instances` para realizar dicha tarea. Algo de suma importancia es que los datos utilizados para generar los últimos conjutos deben ser distintos a los que se usarán posteriormente en el entrenamiento o validación de los modelos. Esto se debe a que si no, se puede generar una dependencia de los datos y podría llevar al sobreajuste (*overfitting*)[1]. El *script* que se debe ejecutar para hacerlo es `tune_params.py`. Su documentación de uso es:

```
$ python tunning/tune_params.py --help

Tune parameters for given models.

Usage:
  tune_params.py --model <name>

Options:
  --model <name>         model name to tune params.
                         Options: svr, rdmforest, pcardmforest, dtr, knnr,
                         mlpr, svr, pcaknnr, pcadtr.
                         If you want to tune all the models together, just
                         put on this parameter 'all'.
  --help                 show this screen
```

Por último, en la Figura **A-1** se puede observar la estructura general del proyecto.

[1]Una analogía clara es que el modelo aprende "de memoria"los datos en vez de comprenderlos. Esto lleva a una muy pobre capacidad de generalización.

```
.
|-- data
|   |-- constants.py
|   |-- data_cleaner.py
|   |-- raw_data
|   `-- scripts
|       `-- clean_data.py
|-- LICENSE
|-- models
|   |-- models.py
|   `-- scripts
|       |-- eval.py
|       |-- plotdata.py
|       `-- train.py
|-- README.md
|-- setup.py
|-- tunning
|   |-- model_tunner.py
|   |-- parameters
|   |   |-- dtr
|   |   |-- knnr
|   |   |-- mlpr
|   |   |-- pcadtr
|   |   |-- pcaknnr
|   |   |-- pcardmforest
|   |   |-- rdmforest
|   |   `-- svr
|   |-- target-runner.py
|   `-- tune_params.py
`-- utils
    `-- utils.py
```

Figura A-1.: Sistema para el ajuste de parámetros y modelado

chapter*

Bibliografía

[1] M. Altamiranda-Saavedra, J. E. Conn, and M. M. Correa. Genetic structure and phenotypic variation of anopheles darlingi in northwest colombia. *Infection, Genetics and Evolution*, 56:143–151, 2017.

[2] V. Andreo, C. Provensal, M. Scavuzzo, M. Lamfri, and J. Polop. Environmental factors and population fluctuations of akodon azarae (muridae: Sigmodontinae) in central argentina. *Austral Ecology*, 34(2):132–142, 2009.

[3] S. Arboleda, N. Jaramillo-O, and A. T. Peterson. Spatial and temporal dynamics of aedes aegypti larval sites in bello, colombia. *Journal of Vector Ecology*, 37(1):37–48, 2012.

[4] D. Basak, S. Pal, and D. C. Patranabis. Support vector regression. *Neural Information Processing-Letters and Reviews*, 11(10):203–224, 2007.

[5] B. E. Boser, I. M. Guyon, and V. N. Vapnik. A training algorithm for optimal margin classifiers. In *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, COLT '92, pages 144–152, New York, NY, USA, 1992. ACM.

[6] L. R. Bowman, G. S. Tejeda, G. E. Coelho, L. H. Sulaiman, B. S. Gill, P. J. McCall, P. L. Olliaro, S. R. Ranzinger, L. C. Quang, R. S. Ramm, et al. Alarm variables for dengue outbreaks: A multi-centre study in asia and latin america. *PLoS One*, 11(6):e0157971, 2016.

[7] L. Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[8] A. L. Buczak, P. T. Koshute, S. M. Babin, B. H. Feighner, and S. H. Lewis. A data-driven epidemiological prediction method for dengue outbreaks using local and remote sensing data. *BMC medical informatics and decision making*, 12(1):124, 2012.

[9] B. Butt, M. D. Turner, A. Singh, and L. Brottem. Use of modis ndvi to evaluate changing latitudinal gradients of rangeland phenology in sudano-sahelian west africa. *Remote Sensing of Environment*, 115(12):3367–3376, 2011.

[10] C. Cardellino, S. Villata, L. Alemany, and E. Cabrio. Information extraction with active learning: A case study in legal text. *Lecture Notes in Computer Science (including*

subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 9042:483–494, 2015.

[11] E. Carranza and A. Laborte. Data-driven predictive modeling of mineral prospectivity using random forests: A case study in catanduanes island (philippines). *Natural Resources Research*, 25(1):35–50, 2016.

[12] R. Caruana and A. Niculescu-Mizil. An empirical comparison of supervised learning algorithms. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, pages 161–168, New York, NY, USA, 2006. ACM.

[13] O. Chapelle, B. Scholkopf, and A. Zien. Semi-supervised learning (chapelle, o. et al., eds.; 2006)[book reviews]. *IEEE Transactions on Neural Networks*, 20(3):542–542, 2009.

[14] S. Cramer, M. Kampouridis, A. A. Freitas, and A. K. Alexandridis. An extensive evaluation of seven machine learning methods for rainfall prediction in weather derivatives. *Expert Systems with Applications*, 85:169–181, 2017.

[15] H. Drucker, C. J. Burges, L. Kaufman, A. J. Smola, and V. Vapnik. Support vector regression machines. In *Advances in neural information processing systems*, pages 155–161, 1997.

[16] M. Espinosa, E. Alvarez Di Fino, M. Abril, M. Lanfri, M. Periago, and C. Scavuzzo. Operational satellite based temporal modeling of aedes population. *Geospatial Health*, 2018.

[17] M. Espinosa, D. Weinberg, C. H. Rotela, F. Polop, M. Abril, and C. M. Scavuzzo. Temporal dynamics and spatial patterns of aedes aegypti breeding sites, in the context of a dengue control program in tartagal (salta province, argentina). *PLoS neglected tropical diseases*, 10(5):e0004621, 2016.

[18] E. L. Estallo, E. M. Benitez, M. A. Lanfri, C. M. Scavuzzo, and W. R. Almirón. Modis environmental data to assess chikungunya, dengue, and zika diseases through aedes (stegomia) aegypti oviposition activity estimation. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 9(12):5461–5466, 2016.

[19] E. L. Estallo, A. E. Carbajo, M. G. Grech, M. Frias-Cespedes, L. López, M. Lanfri, F. F. Ludueña-Almeida, and W. R. Almirón. Spatio-temporal dynamics of dengue 2009 outbreak in córdoba city, argentina. *Acta tropica*, 136:129–136, 2014.

[20] E. L. Estallo, M. A. Lamfri, C. M. Scavuzzo, F. F. L. Almeida, M. V. Introini, M. Zaidenberg, and W. R. Almirón. Models for predicting aedes aegypti larval indices based on satellite images and climatic variables. *Journal of the American Mosquito Control Association*, 24(3):368–376, 2008.

[21] E. L. Estallo, F. F. Luduena-Almeida, A. M. Visintin, C. M. Scavuzzo, M. V. Introini, M. Zaidenberg, and W. R. Almirón. Prevention of dengue outbreaks through aedes aegypti oviposition activity forecasting method. *Vector-Borne and Zoonotic Diseases*, 11(5):543–549, 2011.

[22] E. L. Estallo, F. F. Ludueña-Almeida, A. M. Visintin, C. M. Scavuzzo, M. A. Lamfri, M. V. Introini, M. Zaidenberg, and W. R. Almirón. Effectiveness of normalized difference water index in modelling aedes aegypti house index. *International journal of remote sensing*, 33(13):4254–4265, 2012.

[23] R. Q. Facundo, S. P. Gonzalo, and M. A. Lanfri. Diseño y desarrollo de tecnología para la estratificación de riesgo de circulación viral de dengue a nivel urbano. Tesis de Grado, 2012.

[24] O. Farber and R. Kadmon. Assessment of alternative approaches for bioclimatic modeling with special emphasis on the mahalanobis distance. *Ecological modelling*, 160(1-2):115–130, 2003.

[25] D. O. Fuller, A. Troyo, O. Calderon-Arguedas, and J. C. Beier. Dengue vector (aedes aegypti) larval habitats in an urban environment of costa rica analysed with aster and quickbird imagery. *International Journal of Remote Sensing*, 31(1):3–11, 2010.

[26] B.-C. Gao. Ndwi—a normalized difference water index for remote sensing of vegetation liquid water from space. *Remote sensing of environment*, 58(3):257–266, 1996.

[27] A. German, M. Espinosa, M. Abril, and C. Scavuzzo. Exploring satellite based temporal forecast modelling of aedes aegypti oviposition from an operational perspective. *Remote Sensing Applications: Society and Environment*, 11:231–240, 2018.

[28] A. S. Goldberger et al. Econometric theory. *Econometric theory.*, 1964.

[29] A. d. C. Gomes. Medidas dos níveis de infestação urbana para aedes (stegomyia) aegypti e aedes (stegomyia) albopictus em programa de vigilância entomológica. *Informe epidemiológico do SUS*, 7(3):49–57, 1998.

[30] S. P. Gonzalo and M. A. Lanfri. Geomática aplicada a un sistema de alerta temprana. Tesis de Maestría, 2011.

[31] G. Guo, Y. Fu, C. Dyer, and T. Huang. Image-based human age estimation by manifold learning and locally adjusted robust regression. *IEEE Transactions on Image Processing*, 17(7):1178–1188, 2008.

[32] Y. Guzman-Tapia, M. Ramirez-Sierra, and E. Dumonteil. Urban infestation by triatoma dimidiata in the city of mérida, yucatan, mexico. *Vector-Borne and Zoonotic Diseases*, 7(4):597–606, 2007.

[33] T. Hastie, R. Tibshirani, and J. Friedman. Unsupervised learning. In *The elements of statistical learning*, pages 485–585. Springer, 2009.

[34] S. Hay. An overview of remote sensing and geodesy for epidemiology and public health application. *Advances in parasitology*, 47:1–35, 2000.

[35] S. Hay, M. Packer, and D. Rogers. Review article the impact of remote sensing on the study and control of invertebrate intermediate hosts and vectors for disease. *International Journal of Remote Sensing*, 18(14):2899–2930, 1997.

[36] R. Hecht-Nielsen. Theory of the backpropagation neural network. In *Neural networks for perception*, pages 65–93. Elsevier, 1992.

[37] V. Herbreteau, G. Salem, M. Souris, J.-P. Hugot, and J.-P. Gonzalez. Thirty years of use and improvement of remote sensing, applied to epidemiology: from early promises to lasting frustration. *Health & Place*, 13(2):400–403, 2007.

[38] R. J. Hijmans, S. E. Cameron, J. L. Parra, P. G. Jones, and A. Jarvis. Very high resolution interpolated climate surfaces for global land areas. *International journal of climatology*, 25(15):1965–1978, 2005.

[39] A. H. Hirzel and R. Arlettaz. Modeling habitat suitability for complex species distributions by environmental-distance geometric mean. *Environmental management*, 32(5):614–623, 2003.

[40] A. E. Hoerl and R. W. Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970.

[41] I. Jolliffe. Principal component analysis. In *International encyclopedia of statistical science*, pages 1094–1096. Springer, 2011.

[42] E. Jones, T. Oliphant, P. Peterson, et al. SciPy: Open source scientific tools for Python, 2001–. [Online; accessed ¡today¿].

[43] S. Kalluri, P. Gilruth, D. Rogers, and M. Szczur. Surveillance of arthropod vector-borne infectious diseases using remote sensing techniques: a review. *PLoS pathogens*, 3(10):e116, 2007.

[44] S. Kalluri, P. Gilruth, D. Rogers, and M. Szczur. Surveillance of arthropod vector-borne infectious diseases using remote sensing techniques: a review. *PLoS pathogens*, 3(10):e116, 2007.

[45] B. R. Krasnov, D. Mouillot, G. I. Shenbrot, I. S. Khokhlova, M. V. Vinarski, N. P. Korallo-Vinarskaya, and R. Poulin. Similarity in ectoparasite faunas of palaearctic rodents as a function of host phylogenetic, geographic or environmental distances: which matters the most? *International journal for parasitology*, 40(7):807–817, 2010.

[46] A. Krogh and J. A. Hertz. A simple weight decay can improve generalization. In *Advances in neural information processing systems*, pages 950–957, 1992.

[47] P. Kumar Ra, M. S. Nathawat, and M. Onagh. Application of multiple linear regression model through gis and remote sensing for malaria mapping in varanasi district, india. 2014.

[48] C. Kummerow, W. Barnes, T. Kozu, J. Shiue, and J. Simpson. The tropical rainfall measuring mission (trmm) sensor package. *Journal of atmospheric and oceanic technology*, 15(3):809–817, 1998.

[49] D. Lary, A. Alavi, A. Gandomi, and A. Walker. Machine learning in geosciences and remote sensing. *Geoscience Frontiers*, 7(1):3–10, 2016.

[50] G. Liang, X. Gao, and E. A. Gould. Factors responsible for the emergence of arboviruses; strategies, challenges and limitations for their control. *Emerging microbes & infections*, 4(3):e18, 2015.

[51] M. López-Ibáñez, J. Dubois-Lacoste, L. P. Cáceres, M. Birattari, and T. Stützle. The irace package: Iterated racing for automatic algorithm configuration. *Operations Research Perspectives*, 3:43–58, 2016.

[52] R. K. Meentemeyer, S. E. Haas, and T. Václavík. Landscape epidemiology of emerging infectious diseases in natural and human-altered ecosystems. *Annual review of Phytopathology*, 50:379–402, 2012.

[53] M. Mendez, H. C. Rosenbaum, A. Subramaniam, C. Yackulic, and P. Bordino. Isolation by environmental distance in mobile marine species: molecular ecology of franciscana dolphins at their southern range. *Molecular Ecology*, 19(11):2212–2228, 2010.

[54] N. Metropolis and S. Ulam. The monte carlo method. *Journal of the American statistical association*, 44(247):335–341, 1949.

[55] E. Mirta, V. I. María, and R. Carlos. Directrices para la prevención y control de aedes aegypti. Tesis de Grado, 2012.

[56] T. M. Mitchell. *Machine Learning*. McGraw-Hill, Inc., New York, NY, USA, 1 edition, 1997.

[57] A. C. Moncayo, Z. Fernandez, D. Ortiz, M. Diallo, A. Sall, S. Hartman, C. T. Davis, L. Coffey, C. C. Mathiot, R. B. Tesh, et al. Dengue emergence and adaptation to peridomestic mosquitoes. *Emerging infectious diseases*, 10(10):1790, 2004.

[58] D. R. Montello. The perception and cognition of environmental distance: Direct sources of information. In *International Conference on Spatial Information Theory*, pages 297–311. Springer, 1997.

[59] M. J. Moreno-Madriñán, W. L. Crosson, L. Eisen, S. M. Estes, M. G. Estes Jr, M. Hayden, S. N. Hemmings, D. E. Irwin, S. Lozano-Fuentes, A. J. Monaghan, et al. Correlating remote sensing data with the abundance of pupae of the dengue virus mosquito vector, aedes aegypti, in central mexico. *ISPRS International Journal of Geo-Information*, 3(2):732–749, 2014.

[60] T. Murdoch and A. Detsky. The inevitable application of big data to health care. *JAMA - Journal of the American Medical Association*, 309(13):1351–1352, 2013.

[61] J. Nazzal, I. M. El-Emary, and S. A. Najim. Multilayer perceptron neural network (mlps) for analyzing the properties of jordan oil shale. 5, 01 2008.

[62] A. Y. Ng. Feature selection, l 1 vs. l 2 regularization, and rotational invariance. In *Proceedings of the twenty-first international conference on Machine learning*, page 78. ACM, 2004.

[63] D. Nguyen, R. Gravel, D. Trieschnigg, and T. Meder. "how old do you think i am?.ª study of language and age in twitter. In *ICWSM*, 2013.

[64] R. S. Ostfeld, G. E. Glass, and F. Keesing. Spatial epidemiology: an emerging (or re-emerging) discipline. *Trends in ecology & evolution*, 20(6):328–336, 2005.

[65] G. J. Parra-Henao. Sistemas de información geográfica y sensores remotos. aplicaciones en enfermedades transmitidas por vectores. *CES Medicina*, 24(2), 2010.

[66] E. N. Pavlovsky, P. FK Jr, et al. Natural nidality of transmissible diseases, with special reference to the landscape epidemiology of zooanthroponoses. *Natural nidality of transmissible diseases, with special reference to the landscape epidemiology of zooanthroponoses.*, 1966.

[67] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[68] L. F. Peres and C. C. DaCamara. Land surface temperature and emissivity estimation based on the two-temperature method: Sensitivity analysis using simulated msg/seviri data. *Remote Sensing of Environment*, 91(3-4):377–389, 2004.

[69] R. R. Picard and R. D. Cook. Cross-validation of regression models. *Journal of the American Statistical Association*, 79(387):575–583, 1984.

[70] R. Plamondon and S. Srihari. On-line and off-line handwriting recognition: A comprehensive survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(1):63–84, 2000.

[71] X. Porcasi, C. H. Rotela, M. V. Introini, N. Frutos, S. Lanfri, G. Peralta, E. A. De Elia, M. A. Lanfri, and C. M. Scavuzzo. An operative dengue risk stratification system in argentina based on geospatial technology. *Geospatial health*, 6(3):31–42, 2012.

[72] S. Pradhan, K. Hacioglu, V. Krugler, W. Ward, J. Martin, and D. Jurafsky. Support vector learning for semantic argument classification. *Machine Learning*, 60(1-3):11–39, 2005.

[73] R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2008. ISBN 3-900051-07-0.

[74] W. K. Reisen. Landscape epidemiology of vector-borne diseases. *Annual review of entomology*, 55:461–483, 2010.

[75] E. Rosten and T. Drummond. Machine learning for high-speed corner detection. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 3951 LNCS:430–443, 2006.

[76] C. Rotela, F. Fouque, M. Lamfri, P. Sabatier, V. Introini, M. Zaidenberg, and C. Scavuzzo. Space–time analysis of the dengue spreading dynamics in the 2004 tartagal outbreak, northern argentina. *Acta tropica*, 103(1):1–13, 2007.

[77] C. Rotela, F. Fouque, M. Lamfri, P. Sabatier, V. Introini, M. Zaidenberg, and C. Scavuzzo. Space–time analysis of the dengue spreading dynamics in the 2004 tartagal outbreak, northern argentina. *Acta tropica*, 103(1):1–13, 2007.

[78] C. Rotela, L. Lopez, M. F. Céspedes, G. Barbas, A. Lighezzolo, X. Porcasi, M. A. Lanfri, C. M. Scavuzzo, and D. E. Gorla. Analytical report of the 2016 dengue outbreak in córdoba city, argentina. *Geospatial health*, 12(2), 2017.

[79] H. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(1):23–38, 1998.

[80] D. W. Ruck, S. K. Rogers, M. Kabrisky, M. E. Oxley, and B. W. Suter. The multilayer perceptron as an approximation to a bayes optimal discriminant function. *IEEE Transactions on Neural Networks*, 1(4):296–298, Dec 1990.

[81] L. Rueda, K. Patel, R. Axtell, and R. Stinner. Temperature-dependent development and survival rates of culex quinquefasciatus and aedes aegypti (diptera: Culicidae). *Journal of medical entomology*, 27(5):892–898, 1990.

[82] S. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*. Prentice Hall Press, Upper Saddle River, NJ, USA, 3rd edition, 2009.

[83] S. Salcedo-Sanz, C. Casanova-Mateo, J. Muñoz-Marí, and G. Camps-Valls. Prediction of daily global solar irradiation using temporal gaussian processes. *IEEE Geoscience and Remote Sensing Letters*, 11(11):1936–1940, 2014.

[84] J. Scavuzzo, M. Espinosa, E. Di Fino, M. Abril, G. Peralta, and C. Scavuzzo. Generalización espacial de modelos epidemiológicos basada en el concepto de distancia ambiental normalizada ned. 2018.

[85] J. Scavuzzo, F. Trucco, C. Tauro, A. German, M. Espinosa, and M. Abril. Modeling the temporal pattern of dengue, chicungunya and zika vector using satellite data and neural networks. volume 2017-January, pages 1–6, 2017.

[86] J. M. Scavuzzo, F. Trucco, M. Espinosa, C. B. Tauro, M. Abril, C. M. Scavuzzo, and A. C. Frery. Modeling dengue vector population using remotely sensed data and machine learning. *Acta tropica*, 185:167–175, 2018.

[87] P. Sedgwick. Pearson's correlation coefficient. *Bmj*, 345:e4483, 2012.

[88] A. J. Smola and B. Schölkopf. A tutorial on support vector regression, 2004.

[89] S. Subramanian, S. Huq, T. Yatsunenko, R. Haque, M. Mahfuz, M. Alam, A. Benezra, J. Destefano, M. Meier, B. Muegge, M. Barratt, L. VanArendonk, Q. Zhang, M. Province, W. Petri Jr., T. Ahmed, and J. Gordon. Persistent gut microbiota immaturity in malnourished bangladeshi children. *Nature*, 510(7505):417–421, 2014.

[90] A. J. Tatem, S. J. Goetz, and S. I. Hay. Terra and aqua: new data for epidemiology and public health. *International Journal of Applied Earth Observation and Geoinformation*, 6(1):33–46, 2004.

[91] A. J. Tatem, R. W. Snow, and S. I. Hay. Mapping the environmental coverage of the indepth demographic surveillance system network in rural africa. *Tropical Medicine & International Health*, 11(8):1318–1326, 2006.

[92] P. D. Taylor, L. Fahrig, K. Henein, and G. Merriam. Connectivity is a vital element of landscape structure. *Oikos*, pages 571–573, 1993.

[93] P. F. Vasconcelos and C. H. Calisher. Emergence of human arboviral diseases in the americas, 2000–2016. *Vector-Borne and Zoonotic Diseases*, 16(5):295–301, 2016.

[94] Z. Wan. MODIS land-surface temperature algorithm theoretical basis document (LST ATBD). *Institute of Computational Earth System Science*, 1999.

[95] Z. Wan, Y. Zhang, Q. Zhang, and Z.-L. Li. Quality assessment and validation of the modis global land surface temperature. *International journal of remote sensing*, 25(1):261–274, 2004.

[96] M. Xu, P. Watanachaturaporn, P. K. Varshney, and M. K. Arora. Decision tree regression for soft classification of remote sensing data. *Remote Sensing of Environment*, 97(3):322 – 336, 2005.

[97] D. Zeevi, T. Korem, N. Zmora, D. Israeli, D. Rothschild, A. Weinberger, O. Ben-Yacov, D. Lador, T. Avnit-Sagi, M. Lotan-Pompan, J. Suez, J. A. Mahdi, E. Matot, G. Malka, N. Kosower, M. Rein, G. Zilberman-Schapira, L. Dohnalová, M. Pevsner-Fischer, R. Bikovsky, Z. Halpern, E. Elinav, and E. Segal. Personalized nutrition by prediction of glycemic responses. *Cell*, 163(5):1079 – 1094, 2015.