Uso de Machine Learning para predicciones de COVID-19 en Perú y Arequipa

Alexander Pinto De la Gala
Universidad Católica San Pablo
alexander.pinto@ucsp.edu.pe

Abstract

El presente trata sobre el uso de modelos de Machine Learning para la predicción tomando como caso de uso la actual pandemia de COVID 19 en el Perú y Arequipa. Este trabajo es parte de la evaluación del curso de Tópicos de Inteligencia Artificial de la Universidad Católica San Pablo de Arequipa, Perú.

1. Introduction

El presente documento trata sobre el uso de modelos de Machine Learning para la predicción de variables, tomando como caso de uso la la actual pandemia de COVID 19 en el Perú y Arequipa para la predicción de número de casos y número de muertes. La enfermedad COVID 19 tuvo su primer caso detectado en el Perú, el 06 de marzo del presente año. A la fecha 22 de mayo de 2020 se cuenta un total de 111 698 casos confirmados y un total de 3244 fallecidos [1].

El área de Machine learning ofrece diferentes herramientas para la predicción, para el presente trabajo se ha tomado los modelos de aprendizaje supervisado, Suport Vector Machine (SVM), Regresión Polinomial (RP) y Regresión Lineal Múltiple (RLM).

Este manuscrito está dividido de la siguiente forma: en la Sección 2 Marco Teórico, haremos una breve introducción sobre la teoría acerca de los métodos utilizados; en la Sección 3 Experimentos, trataremos sobre el proceso de la experimentación y finalmente en la Sección 4 Resultados y Conclusiones, haremos una discusión sobre los resultados obtenidos de cada experimento y resumiremos los detalles más importantes del presente trabajo.

2. Marco Teórico

2.1. Support Vector Machine (SVM)

SVM es usualmente considerado un modelo de clasificación, pero es empleado también en modelos de regresión. Puede utilizar tanto variables continuas como categóricas. SVM construye un hiperplano en un espacio multidimensional para separar diferentes clases. Los *support vectors* son definidos como los puntos de datos más cercanos que recaen a la superficie de decisión o hiperplano (R. Berwick, 2003). Estos son los puntos más difíciles de clasificar y tienen influencia directa en la ubicación óptima de la superficie de decisión. En general existen infinitas soluciones para ubicar el hiperplano.

El margen es un brecha (*gap*) entre las dos líneas sobre los puntos más cercanos de las diferentes clases. El margen es calculado como la distancia perpendicular desde la línea que conforma los vectores de soporte. Si el margen es mayor entre las clases se considera un buen margen, si es más pequeño es un mal margen.

El algoritmo de SVM puede expresarse como: 1. Generar hiperplanos los cuales segregan las clases de la mejor manera; 2. Seleccionar el hiperplano correcto con la máxima segregación desde sus puntos más cercanos; y 3. Repetir hasta convergencia o número de iteraciones.

Debido a que no todos los problemas pueden resolverse utilizando un hiperplano lineal, SVM utiliza lo que se conoce como *kernel trick*, lo cual es una transformación del espacio de entrada un espacio

[1] Sala Situacional - MINSA

dimensional mayor, de manera que se pueda aplicar una separación lineal. Las ecuaciones 1 tratan esta transformación.

$$\text{Kernel Lineal :} \quad K(x, x_i) = \sum(x \cdot x_i) \tag{1}$$

$$\text{Kernel Polinomal :} \quad K(x, x_i) = 1 + \sum(x \cdot x_i)^d \tag{2}$$

$$\text{Kernel Función Radial :} \quad K(x, x_i) = \exp(\gamma * \sum(x - x_i^2) \tag{3}$$

donde d es el grado del polinomio y $gamma$ es un parámetro entre 0 y 1 el cual tiene que ser ajustado manualmente.

2.2. Regresión Lineal Múltiple (RLM)

El modelo RLM parte de una hipótesis Ec. (4) que aproxima y a una función lineal de x:

$$h_\theta(x) = \theta^T x = \theta_0 x_0 + \theta_1 x_1 + ... + \theta_n x_n \tag{4}$$

donde $\theta_0, \theta_1, \theta_2 ... \theta_n$ son los parámetros, también llamados pesos, los cuales parametrizan el espacio de la función lineal mapeando de X a Y.
La función de costo a minimizar está definida por:

$$J(\theta) = \frac{1}{2m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)})^2 \tag{5}$$

El algoritmo de Gradiente descendiente se representa como:
Repetir:

$$\theta_j = \theta_j - \alpha \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)}) x_j^{(i)} \tag{6}$$

donde: α es la tasa de aprendizaje.

La resolución analítica de θ está dada por la Ecuación Normal:

$$\theta = (X^T X)^{-1} X^T y \tag{7}$$

2.3. Regresión Polinomial (RP)

RP es técnicamente un caso especial de RLM donde la hipótesis toma la forma de:

$$h_\theta(x) = \theta^T x = \theta_0 x_0 + \theta_1 x_1 + \theta_2 x_2^2 + \theta_3 x_3^3 + ... + \theta_n x_n^m \tag{8}$$

Presenta algunas ventajas frente a RLM como: Provee una mejor aproximación de las relaciones entre la variable dependiente e independiente, un mayor rango de funciones pueden ser modeladas, y un polinomio puede ajustarse a un mayor rango de curvaturas. Sin embargo, tiene algunas desventajas como la presencia de uno o más *outliers* en los datos afecta seriamente los resultados de un análisis no lineal y existen menos modelos de validación para la detección de *outliers* que la regresión lineal.

3. Experimentos

En esta sección trataremos sobre la aplicación de estos tres modelos para la predicción de casos y muertes por COVID 19 en Perú y Arequipa [2]. Se tomaron en cuenta algunos lineamientos utilizados en diversos notebooks de *kaggle*[3]. Los experimentos fueron realizados con Python 3.7 utilizando las bibliotecas pandas y numpy para el manejo de datos, sklearn para los modelos de SVM y RP; para el caso de RLM se utilizó una implementación propia; para el ploteo de gráficos se utilizó matplotlib.

[2]Código de la implementación disponible en https://github.com/giulianodelagala/COVID19PeruArequipa/
[3]https://www.kaggle.com/therealcyberlord

3.1. Extracción y Preprocesamiento de Datos

Los datos utilizados para los experimentos son publicados por la Sala Situacional del Ministerio de Salud (MINSA) y recopilados por Jesús M. Castagnetto [4]. El *dataset* contiene información del conteo diario acumulado de casos confirmados, muertes y recuperados. También existe datos sobre casos negativos y el tipo de prueba realizada, sin embargo es data redundante con respecto al número de casos. Cuenta con un total de 1500 registros al 18 de mayo de 2020. La data se encuentra agrupada por regiones, sin embargo la información es incompleta para algunas regiones, o presenta outliers (ver Fig. 1). Solo se utilizó las columnas correspondientes a casos confirmados (*confirmed*), muertes (*deaths*) y recuperados (*recovered*). Los datos incompletos fueron rellenados por el valor cero.

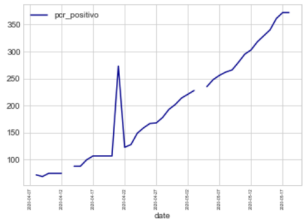


Figura 1: Presencia de outliers y data incompleta

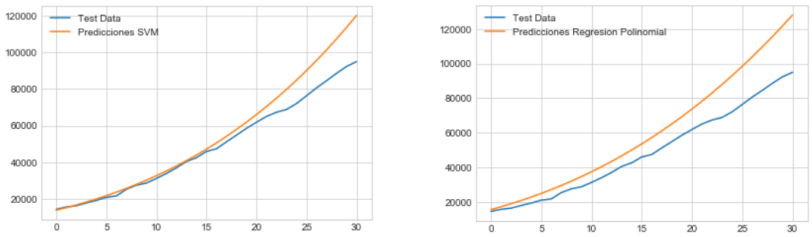
La data fue dividida en dos subconjuntos para el entrenamiento y pruebas, tomando un 42 % del tamaño del dataset para las pruebas.

3.2. Experimentos

En el caso de los modelos SVM y RP, la Tabla 1, resume los errores MAE y MSE. La Figura 2, muestra la comparación del resultado del entrenamiento contra el subconjunto de prueba en dataset Perú.

Tabla 1: Errores para SVM y RP

Casos	SVM	RP
MAE	5780.30	11506.55
MSE	83177956.43	211205619.35
Muertes	SVM	RP
MAE	73.96	317.56
MSE	11532.28	143359.38



(a) SVM (b) Reg.Polinomial

Figura 2: Comparativo Modelos versus Test

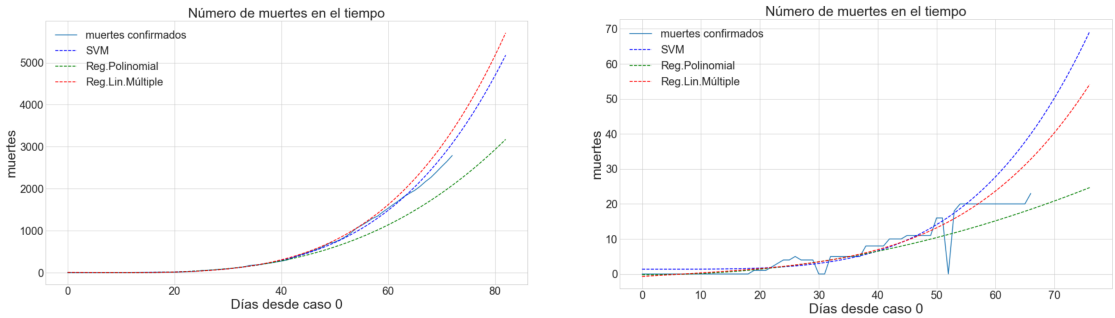
Para el caso de RLM, se tomó como datos de entrada fecha (*date*)y casos confirmados (*confirmed*), debido a que la predicción requiere como entrada estos mismos datos, se tomó los resultados dados por SVM para casos confirmados, sólo predecimos número de fallecidos (death) para este modelo.

Los resultados de la predicción fueron obtenidos luego de un refinamiento de los parámetros de entrada. La Tabla 2 resume estas predicciones para Perú y Arequipa. En la Fig. 3 se presentan las predicciones hasta el 27 de mayo de 2020.

[4]https://github.com/jmcastagnetto/covid-19-peru-data/tree/master/datos

Tabla 2: Predicciones de Casos y Muertes para Perú y Arequipa

Perú						Arequipa				
	SVM		Reg.Polinomial		Reg.Lineal Múltiple	SVM		Reg.Polinomial		Reg.Lineal Múltiple
Fecha	Casos	Muertes	Casos	Muertes	Muertes	Casos	Muertes	Casos	Muertes	Muertes
05/22	157088	4026	163136	2588	4432	3335	73	795	25	57
05/23	165408	4239	170844	2699	4668	3512	77	817	26	60
05/24	174054	4461	178791	2813	4914	3695	81	840	27	62
05/25	183035	4691	186980	2930	5169	3885	85	863	27	65
05/26	192359	4930	195414	3050	5434	4082	89	886	28	68
05/27	202035	5178	204098	3174	5709	4287	93	910	29	72



(a) Perú (b) Arequipa

Figura 3: Predicción de número de muertes

Finalmente en la Figura 4 presentamos la variación de la tasa de mortalidad en el caso de Perú.

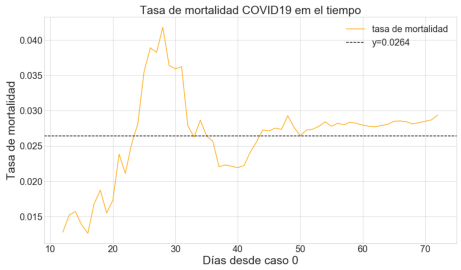


Figura 4: Variación de la Mortalidad en Perú

4. Resultados y Conclusiones

Para el caso del comparativo entre SVM y RP, podemos ver que SVM presenta errores menores tanto para el caso de predicción de casos y muertes. En ambos modelos se utilizó solo una variable de entrada. Sin embargo si comparamos con datos al 22 de mayo, los casos actuales confirmados (111698) y muertes (3242), vemos que SVM tiene un error por exceso. Este comportamiento indica que la tendencia de la curva ha cambiado por lo que en cierta medida las medidas de control de propagación han tenido un efecto. Para el caso de Arequipa se tienen confirmados 2373 casos y 42 muertes, SVM tiene también un error por exceso (3335), RP da valores muy lejanos probablemente por la presencia de outliers. La RLM tiene mejores resultados para el pronóstico de muertes (57). En cuanto a tasa de mortalidad vemos que se presenta un promedio de 2.6 % a nivel nacional. Se debe entender que los datos oficiales pueden presentar error debido a la propia naturaleza de la toma de muestras y control de registros.

References

Andrew Ng, CS229 Lecture Notes

R. Berwick, An Idiot's guide to Support Vector Machines (SVMs) 2003