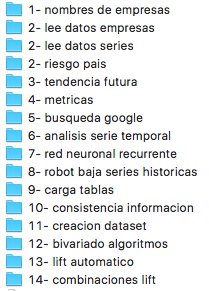
Detalle de algoritmos en proyecto de tesis

Herramientas utilizadas: Python, R, SQL, Excel, Sublime.



1-nombres empresas

Se obtiene los nombres de las empresas de la bolsa de ámbito.com y se copian al csv **nombre\_empresas.csv**

2- lee datos empresas

En **scraping.py** se realiza una bajada de información de cotizaciones de la página web.

2- lee datos series

En **scraping.py** se leen datos de series del banco central para analizar correlación con cotizaciones (tasas de interés, reservas monetarias, bonos).

En **metricas\_bcra.csv** el resultado del dataset.

2- riesgo país

Fuente de datos riesgo país para descargar

3- tendencia futura

En **tendencia.py** se crea una regresión que ajuste a valores futuros para sacar la tendencia en próximos días con el objetivo de obtener un target continuo para analizar.

4- metricas

En creación de **métricas.py** se crean métricas en base a la historia de los cierres.

5- búsqueda google

Se crea un robot que realiza una búsqueda por empresa, utiliza chromedriver para simular el comportamiento humano.

Recupera día a día en el pasado las páginas en las cuales fue nombrada la empresa y por cada página realiza un copy paste de todo lo que encuentre en el display (si captamos solo el código fuente perdemos información valiosa por eso se usa chromedriver).

Una vez copiado el texto se realiza un bag of words con el objetivo de analizar como el contexto social influye en fluctuaciones de las acciones de la empresa. Se eliminan páginas no deseadas realizando macheo por expresiones regulares teniendo en cuenta el contexto de la palabra buscada.

6-Analisis serie temporal

En **descomposicion.R** se descompone la serie temporal en componentes de tendencia, estacionalidad y aleatorio.

**lag\_overPartition.sql** creación de lags y tendencias

En **series.py** grafica series

El objetivo es buscar correlaciones entre series temporales variando el lag de la serie

7- red neuronal recurrente

En “prob\_dataset\_pampa\_petb/**iterador.py”** se crea un iterador de parámetros para entrenar varias redes neuronales recurrentes y quedarse con la que mejor ajusta al conjunto de datos.

8- robot baja series históricas

Bajada histórica de tres fuentes de datos distintas para asegurarse consistencia de la información. Robot que recorre páginas web simulando comportamiento humano con scraping y webdriver.

Por ejemplo en “bajada de invertironline\**acciones\_Arg.py”** se realiza la bajada de información de invertironline.

9- carga tablas

Carga las fuentes descargadas en el punto anterior en una base de datos MySQL. Son varias tablas, hacer la subida manualmente llevaría mucho tiempo por lo que automatice la creación de tablas con Python.

Por ejemplo en “investing/**python crea script carga.py”** por cada fuente de datos correspondiente a cotización de una acción se crea un script correspondiente para la subida a Mysql. Se puede ver el resultado de la query en **salida\_script.txt** lista para ejecutarla en Mysql.

10- consistencia información

Chequeos de consistencia de las fuentes anteriores en consultas SQL.

11- creacion dataset

Crea métricas con nuevo dataset en “**creacion de metricas.py”** observando la historia.

12- bivariado algoritmos

Algoritmo que itera creando combinaciones tomadas de a dos variables. Por cada conjunto de dos variables se les aplica una transformación SVM para encontrar una nueva dimensión que pueda separar el problema (rbf,sigmoid,linear). La nueva variable contiene información de probabilidades que podrán ser tratadas de forma continua en otro algoritmo.

Se puede ver el código de prueba en **“transformacion de variables.py”**

13- lift automatico

En “**Cruce Variables - COBRANZAS.xlsx”** creo una métrica para seleccionar variables teniendo en cuenta el lift en cada subconjunto respecto al share del total para evaluar la fuerza de discriminación.

En **lift.py** se replica la lógica para hacerlo automático.

14- combinaciones lift

El objetivo de este script es encontrar conjuntos que representen un lift superior al conjunto total para identificar grupos de acción, desvíos o anomalías.

Si el porcentaje de morosidad aumenta y no sabemos a qué se debe. El objetivo de este algoritmo es realizar combinaciones de variables y calcula la mora en cada subconjunto para identificar si hay crecimiento o desvíos respecto de la métrica global

En **prueba\_transformaco5.py** se pueden ver las pruebas, hay problemas de performance por cantidad de variables.

15- transformacion correlacion

En el código “**en masa11.py**”se realizan un análisis por correlaciones para eliminar variales con poca información cuando contamos con muchas variables.

A las variables categóricas se les aplica una transformación por diferencia de medias para que puedan ajustar con información del target. También se les realiza una transformación creando variables dummies.

Teniendo estas transformaciones de variables categóricas y sumando las continuas estamos en condiciones de aplicar correlaciones respecto al target. Si el valor absoluto de la correlación respecto al target es mayor que 0.2 se guarda la variable en una lista.

Luego con todas las variables en la lista se busca la correlación entre ellas mismas para eliminar las colineales. Luego de este proceso tenemos un conjunto de variables para una regresión evitando colinealidad.