Tarea 1 - EDA y modelos bayesianos

MA6202 Laboratorio de Ciencia de Datos

Otoño 2020

1 Introducción

La siguiente evaluación corresponde a la primera tarea del curso de laboratorio de ciencia de datos. A modo de contexto, se trabajará con un conjunto de datos, en el cual se busca una estimación del precio por metro cuadrado para viviendas en la ciudad de Bogotá, se busca además una estimación de incertidumbre para sus predicciones. Para ello, se proporcionan datos recolectados por *web-scrapping* y una serie de estadísticas que caracterizan determinadas zonas geográficas de interés, llamadas Unidades de Planeación Zonal (UPZ).

Se evaluará la presentación de sus resultados por medio de un informe, las condiciones de entrega requeridas son:

- La extensión máxima del informe es de 6 planas a las que puede añadir 2 para demostraciones.
- Debe adjuntar un repositorio `git` donde se incluya todo su código.
- A lo menos 1 `commit` por cada pregunta de la tarea
- Por lo menos 1 `merge` a través de su trabajo.
- Incluya un documento `jupyter notebook` llamado `tarea1.ipynb` en el cual se exponga todo el procedimiento realizado.
- Por último es necesario también entregar un archivo *pickle* denominado `modelo.pk` que contenga el último modelo de regresión entrenado.

Tenga en mente que su informe será revisado por un equipo técnico que debe entender a cabalidad su metodología, ser capaz de replicarlo y evaluarlo a partir de su lectura.

P1. Carga y limpieza de datos

En la presente sección se realizan los pasos de carga y limpieza de datos que permitirán realizar las secciones posteriores con un `DataFrame` consolidado que presente los tipos de datos adecuados para la información contenida en sus columnas.

Incluya en el reporte todas las decisiones que llevó en esta sección, además de reportar y discutir acerca de los aspectos específicos señalados en cada pregunta.

1. Los datos recolectados en la carpeta `data/raw` están divididos en carpetas con el formato `wNN` donde `NN` corresponde a la semana del año en la que estos fueron consultados. Cargue los datos en un solo `DataFrame` y elimine las filas duplicadas. Además genere una variable categórica en la que se indique si la observación correspondiente proviene de un archivo que en su nombre contiene `'furnished'` (ejemplo: metrocuadrado_furnished_wNN.csv). *Hint*: para lo último puede ser útil estudiar los argumentos del método `pd.merge`. Reporte si existen observaciones de archivos con texto `'furnished'` que no estén contenidos en archivos con texto `'all'`.
2. Limpie las columnas:
   (a) Precio y de área del inmueble, número de habitaciones y número de baños.

(b) De `'property_type|rent_type|location'`. Debe obtener 3 columnas en las que se detalle el tipo de inmueble (casa o apartamento), el tipo de la oferta (arriendo o arriendo y venta) además de el barrio en el cual se ubica el inmueble (texto en mayúsculas para la mayoría de los casos) Llame a esta última columna `'location'`. *Hint*: Para ello pueden ser útiles los métodos `str.split` y `str.strip` de la clase `pd.Series`.

3. Adjunte las siguientes columnas:

   (a) Genere una variable que represente el precio por metro cuadrado.

   (b) Obtenga el número de garajes procesando la columna `'url'` .

4. Clasifique las observaciones por tipo de producto de acuerdo a los criterios de la Tabla 1. *Hint*: puede ser útil el método `query` de `pd.DataFrame`

tipo de producto	tipo de inmueble	área min	área max
1	casa	80	<120
2	casa	120	<180
3	casa	180	<240
4	casa	240	<360
5	casa	360	460
6	apartamento	40	<60
7	apartamento	60	<80
8	apartamento	80	120

Table 1: Tipos de productos

5. A partir de la columna `'barrio'`, haga una fusión con el archivo `data/assignacion_upz/barrio-upz-asignacion.csv` para obtener así el código de la UPZ de cada inmueble. Reporte el numero de observaciones y de barrios a los que no se les puede adjuntar un código UPZ a partir de este archivo. Tenga en cuenta que aproximadamente 90% de los datos tiene información sobre UPZ.

6. En la carpeta `data/estadisticas_upz` encontrará todos los archivos que debe fusionar con su `DataFrame`, a través del código de UPZ, para así enriquecer su conjunto de datos con estadísticas de población, socioeconómicas y de calidad de vida a nivel de UPZ. Una vez realizada la fusión, adjunte una nueva columna con la densidad de población por UPZ.

P2. EDA

En la siguiente pregunta utilice el `DataFrame` obtenido a partir de los procedimientos anteriores. La idea central del presente ejercicio es analizar la base de datos que fue construida realizando un análisis exploratorio de datos (EDA) riguroso que permite obtener información útil para la parte final de tarea. Para cada pregunta y considerando que la variable de respuesta es el precio por metro cuadrado, discuta sus resultados y reporte algunos gráficos interesantes en su informe.

1. Programe una función `estilo()` que aplica un estilo de gráficos por defecto diseñado por usted. Este estilo debe ser empleado en todos los gráficos que incluya en su informe. *Hint*: Use métodos de la librería `seaborn` para ello.

2. Perfile las variables obtenidas en su `DataFrame`, agrúpelas y analícelas según su naturaleza. En función de su agrupación, grafíque las distribuciones dando un tratamiento adecuado a cada tipo de variable, discuta ciertos casos de interés.

3. Estudie la presencia de datos faltantes en la base de datos. Observe como se distribuyen estos y establezca un mecanismo de pérdida de información basándose en los patrones observables del conjunto de datos. Busque agrupaciones de columnas que muestren un comportamiento sistemático y plantee sus reflexiones. Respalde con visualizaciones y cuantifique estadísticamente los patrones observados.

4. Recategoríce la variable código de UPZ de forma que quede distribuida entre 3 a 5 grupos, evalúe la significancia estadística de esta nueva agrupación en comparación a la variable de respuesta. Comente sus resultados e interprételos.

   Hint: Puede probar con técnicas de clustering (como k-means) sobre una agrupación de UPZ y validar estadísticamente si las nuevas categorías afectan la variable de respuesta.

5. Cuantifique estadísticamente las relaciones entre una selección de al menos 10 variables de interés y la columna de respuesta, examine también las relaciones entre las variables de su selección. Utilice las herramientas de análisis estadístico que considere pertinentes, comente brevemente sus hallazgos.

6. En base a las herramientas del curso realice un análisis que permita detectar observaciones anómalas en la base de datos, justifique sus resultados, evalúe como se distribuyen los valores anómalos respecto a las variables UPZ y tipo de producto.

7. En función del análisis realizado a lo largo de esta pregunta, proponga una selección de variables que permita estimar la variable respuesta por medio de un modelo de regresión, discuta.

P3. Regresión lineal bayesiana

Una vez analizado el conjunto de datos, se procede a modelar las relaciones por medio de alguna herramienta matemática. En este contexto y con las herramientas computacionales entregadas por el curso, se desarrolla un modelo de regresión lineal bayesiana, tratando el problema de modelación desde el punto vista teórico hasta su implementación e interpretación de resultados.

Desarrollo teórico

Dado un conjunto de observaciones $D = \{(\mathbf{x}_i, y_i)\}_{i=1}^N \in \mathbb{R}^{d-1} \times \mathbb{R}$ el modelo de regresión lineal, concibe el problema de aproximación mediante el esquema:

$$y_i = w_0 + \sum_{j=1}^{d-1} w_j x_{i,j} + \epsilon_i = \mathbf{w}^T \mathbf{x}_i + \epsilon_i \tag{1}$$

Con la convención $x_{i,0} = 1 \ \forall i \in \{1, ..., N\}$ y con $\epsilon_i \sim \mathcal{N}(0, \beta^{-1})$ i.i.d $\forall i \in \{1, ..., N\}$. Dada la suposición de normalidad del error, es decir $\epsilon_i \sim \mathcal{N}(0, \beta^{-1})$, la verosimilitud para una observación se expresa según

$$p(y_i|\mathbf{x}_i, \mathbf{w}, \beta) \sim \mathcal{N}(\mathbf{w}^T \mathbf{x}_i, \beta^{-1})$$

Para este problema, la familia de funciones aproximadoras $\mathcal{M}$, denominada como *modelo*, corresponde a las funciones lineales representadas por $\mathbf{w}$. Así, si se reescribe la verosimilitud anterior como $p(\mathcal{D}|\mathcal{M})$, según el teorema de Bayes, se tiene

$$p(\mathcal{M}|\mathcal{D}) = \frac{p(\mathcal{D}|\mathcal{M})p(\mathcal{M})}{p(\mathcal{D})} \tag{2}$$

Es decir, la probabilidad posterior $p(\mathcal{M}|\mathcal{D})$ de elegir una función aproximadora, dado el conjunto de datos, puede ser calculada en función de una probabilidad a priori, sobre el modelo $p(\mathcal{M})$, la verosimilitud $p(\mathcal{D}|\mathcal{M})$ y la evidencia $p(\mathcal{D})$. El enfoque bayesiano, consiste en modelar una probabilidad a priori $p(\mathcal{M})$ para obtener la función que mejor modela los datos utilizando la expresión (2) para trabajar con $p(\mathcal{M}|\mathcal{D})$. Observe que en el caso del modelo lineal (1), obtener una expresión para $p(\mathcal{M})$, es equivalente a modelar una distribución sobre $\mathbf{w}$.

1. Utilice una distribución gaussiana isotrópica sobre los parámetros $\mathbf{w}$ con media cero:

   $$p(\mathbf{w}|\alpha) \sim \mathcal{N}(\mathbf{0}, \alpha^{-1}\mathbf{I})$$

   considere que $p(\mathcal{D})$ es una constante. Con esto, muestre que la distribución posterior de los parámetros $\mathbf{w}$ es proporcional a

   $$p(\mathbf{w}|\mathbf{y}, \mathbf{X}, \alpha, \beta) \sim \mathcal{N}(\mathbf{m}_N, \mathbf{S}_N)$$

   donde

   $$\begin{aligned} \mathbf{m}_N &= \beta \mathbf{S}_N \mathbf{X}^T \mathbf{y} \\ \mathbf{S}_N^{-1} &= \alpha \mathbf{I} + \beta \mathbf{X}^T \mathbf{X} \end{aligned}$$

2. Para hacer predicciones en nuevos puntos $\mathbf{x}'$, se utiliza la probabilidad posterior predictiva, dada por

   $$p(y'|\mathbf{x}', \mathbf{y}, \mathbf{X}, \alpha, \beta) = \int p(y'|\mathbf{x}', \mathbf{w}, \beta) p(\mathbf{w}|\mathbf{y}, \mathbf{X}, \alpha, \beta) d\mathbf{w}$$

Muestre que la distribución predictiva posterior para una observación $\mathbf{x}'$ dados α y β tiene la forma:

$$p(y'|\mathbf{x}', \mathbf{y}, \mathbf{X}, \alpha, \beta) \sim \mathcal{N}(\mathbf{m}_N^T \mathbf{X}, \sigma_N^2(\mathbf{x}'))$$

donde

$$\sigma_N^2(\mathbf{x}') = \frac{1}{\beta} + \mathbf{X}^T \mathbf{S}_N \mathbf{x}'$$

Hint: puede ser útil observar la integral anterior como una convolución de gaussianas.

3. Hacer estimaciones con este modelo requiere de los parámetros α y β. Para estimarlos, se utilizará la información contenida en los datos, esto se denomina *enfoque bayesiano empírico*. Use el teorema de probabilidades totales para deducir que la log-verosimilitud de $p(\mathbf{y}|\alpha, \beta)$ tiene la forma:

$$\log p(\mathbf{y}|\alpha, \beta) = \frac{d}{2} \log \alpha + \frac{N}{2} \log \beta - E(\mathbf{m}_N) - \frac{1}{2} \log |\mathbf{S}_N^{-1}| - \frac{N}{2} \log 2\pi \tag{3}$$

donde

$$E(\mathbf{m}_N) = \frac{\beta}{2} ||\mathbf{y} - \mathbf{X}\mathbf{m}_N||_2^2 + \frac{\alpha}{2} \mathbf{m}_N^T \mathbf{m}_N$$

Hint: es necesario integrar utilizando dos distribuciones enunciadas anteriormente.

4. (Opcional) La función (3) se denomina log-verosimilitud marginal con respecto a α y β. Muestre que al maximizar dicha función se obtienen las soluciones implícitas:

$$\begin{aligned} \alpha &= \frac{\gamma}{\mathbf{m}_N^T \mathbf{m}_N} \\ \frac{1}{\beta} &= \frac{1}{N - \gamma} \sum_{i=1}^{N} (y_i - \mathbf{m}_N^T \mathbf{x}_i)^2 \\ \gamma &= \sum_{i=0}^{M-1} \frac{\lambda_i}{\alpha + \lambda_i} \end{aligned}$$

donde λ_i son los valores propios de $\beta \mathbf{X}^T \mathbf{X}$. Las soluciones están implícitas pues α, β y γ dependen unas de otras.

Observe que las expresiones obtenidas entregan un esquema para aproximar los parámteros óptimos. Para esto, sólo hace falta comenzar con valores iniciales de α y β, con los cuales se calcula γ y se obtiene $\mathbf{m}_N$ de la expresión asociada a la probabilidad posterior $p(\mathbf{w}|\mathbf{y}, \mathbf{X}, \alpha, \beta)$. Esto permite actualizar el valor de α y luego de β, completando un ciclo. El esquema de aproximación se detiene cuando no hay mayores diferencias entre actualizaciones de α y β.

Implementación

1. Implemente la clase `RegresionBayesianaEmpirica` que herede de `BaseEstimator` y de `RegressorMixin` del módulo `sklearn.base` en la cual se implementa la heurística enunciada en la sección anterior para aproximar los hiperparámetros óptimos α y β. Esta clase sólo debe usar objetos de la librería `NumPy` y debe incluir al menos los siguientes métodos:

   - `__init__(self, alpha_0, beta_0, tol=1e-5, maxiter=200)`: sus argumentos son auto explicativos.
   - `get_posteriori(self, X, y, alpha, beta)`: que reciba la matriz de observaciones (`X`), el vector de etiquetas (`y`) y los hiperparámetros α y β. Este debe retornar los objetos necesarios para interactuar con los demás métodos.
   - `fit(self, X, y)`: que reciba la matriz de observaciones (`X`), el vector de etiquetas (`y`) e implemente el esquema de aproximación mencionado. Este método debe guardar como atributos del objeto los parámetros óptimos obtenidos, además de reportar en pantalla indicadores del proceso iterativo (incluya al menos el número de iteraciones).
   - `predict(self, X_, return_std=False)`: que reciba una matriz de observaciones (`X_`). Debe retornar la tupla `(y_, y_std)` con el vector de medias y el de desviaciones estándar (cuando `return_std=True`) asociadas a las observaciones en `X`. Para esto, observe que el proceso de predicción corresponde a asignar la media posterior predictiva del modelo a nuevos puntos.

Note que todo desarrollo sólo necesita de un modelo lineal en los parámetros $\mathbf{w}$. Es decir, es posible reemplazar $\mathbf{X}$ por una transformación (posiblemente no lineal) $\mathbf{\Phi}(X)$, manteniendo la misma estructura distrubucional tanto en predicción como en obtención de hiperparámetros.

En las siguientes preguntas se construye un modelo de regresión donde su variable de respuesta es el precio por metro cuadrado. Su matriz de observaciones debe incluir todas las variables disponibles en el conjunto de datos, exceptuando las columnas de precio, área y tipo de vivienda (casa o departamento), además de reemplazar la variable de UPZ por la recategorización antes propuesta. Adicionalmente debe eliminar las filas que contengan datos faltantes en las columnas: precio por metro cuadrado, tipo de producto y código UPZ.

2. Construya un flujo de transformaciones sobre el conjunto datos. Por medio de la clase `Pipeline` deberá:

   - Utilizar `StandardScaler`, `MinMaxScaler` y `OneHotEncoder` donde corresponda.
   - Utilizar el objeto `PolynomialFeatures` para generar características polinomiales, en este apartado, se recomienda utilizar características de grado 3 sólo en las variables numéricas y luego concatenar con las codificaciones categóricas.
   - Generar una composición de transformaciones por medio de `ColumnTransformer`.

3. Expanda el `Pipeline` anterior agregando el modelo representado en la clase `RegresionBayesianaEmpirica`. Genere una separación de entrenamiento y test por medio de `train_test_split` del módulo `model_selection` donde el 20% de los datos sea de test. Entrene su modelo utilizando el método `.fit` como parámetros dentro del flujo creado por medio de `Pipeline`, evalúe su modelo por medio de la raíz del promedio de errores cuadráticos (*Root Mean Square Error - RMSE.* en inglés) en el conjunto de test, incluya el estadístico R^2.

   Hint: pruebe con valores para `alpha_0` y `beta_0` entre 10^{-10} y 10^{-5}. Compruebe que dichos dichos hiperparametros son adecuados, al obtener un R^2 cercano a 0.7.

4. Utilice su selección de variables transformándolas. Adapte el esquema de preprocesamiento anterior, separe en conjuntos de entrenamiento y test manteniendo el 20% de proporción y evalúe los resultados del modelo `RegresionBayesianaEmpirica` incluyendo el estadístico R^2. Discuta.

5. Finalmente, compare los resultados de su selección de variables con los obtenidos por un pipeline donde el estimador sea uno de la clase `BayesianRidge` del módulo `sklearn.linear_model`. Discuta la diferencia de este módulo con respecto al desarrollado por ustedes.