Los datos los puede encontrar en el siguiente link:  
<https://www.kaggle.com/parulpandey/forbes-highest-paid-athletes-19902019>  
Ustedes tendrán que revisar los datos y realizar lo siguiente:  
1.     Cargue los datos en un dataframe  
2.     Identificar el tipo de datos de cada variable.  
3.     Calcular estadísticas básicas para cada variable: posición, dispersión y tendencia central. Para el caso de variables categóricas calcule la moda y la frecuencia absoluta y frecuencia relativa.  
4.     Realice al menos 4 gráficos con lo desarrollado en el punto anterior. Incluya gráficos con variables numéricas y gráficos con variables categóricas.  
5.     Transforme la variable “earnings” en una variable categórica implementando intervalos de clase. Grafique el resultado.  
6.     Conteste las siguientes preguntas:  
a.- ¿Quién es la persona que más dinero ganó en un año?  
b.- ¿Cuál es el deporte que más se repite dentro de la lista?  
c.- ¿Cuál es el deporte donde más dinero se gana?  
 Debe entregar un Notebook con todo el desarrollo.

**Regresión**  
1. Cargue los datos Life Expectancy Data.csv  
Además, podrá encontrarlo en el siguiente link (use solo el archivo train.csv):  
<https://www.kaggle.com/kumarajarshi/life-expectancy-who>  
  
  
En este archivo usted podrá encontrar información acerca de la expectativa de vida de varios países durante los años 2000 y 2015. En el link encontrará la descripción en detalle de esta información.    
Guarde los datos en un Dataframe.  
En este dataset usted encontrará datos faltantes, variables categóricas y numéricas.   
   
Elija como variable dependiente: “Life Expectancy”, y como variables independientes 5 variables a su elección. Considere estas variables para realizar los siguientes puntos 2, 3 y 4.  
  
2. Realice regresión lineal considerando las variables independientes para estimar la variable dependiente (Life expectancy). Divida en training set y testing set. Obtenga las métricas de rendimiento vistas en clases.  
  
3. Realice regresión Lasso considerando las variables independientes para estimar la variable dependiente (Life expectancy). Divida en training set y testing set. Obtenga las métricas de rendimiento vista en clases. Además, obtenga los coeficientes de regresión y explique cuáles variables fueron consideradas en la regresión y cuáles no.  
  
4. Realice regresión mediante árbol de regresión considerando las variables independientes para estimar la variable dependiente (Life expectancy). Divida en training set y testing set. Obtenga las métricas de rendimiento vistas en clases.  
  
**Clasificación**  
  
1. Elija como variable dependiente: “Status”, y como variables independientes al menos 10 variables a su elección. Considere estas variables para realizar los siguientes puntos 2, 3 y 4. Si fuese necesario, realice: encodeo, normalización, estandarización, imputación, etc.  
  
2. Realice clasificación usando el algoritmo K-NN considerando las variables independientes para estimar la variable dependiente (Status). Divida en training set y testing set. Obtenga las métricas de rendimiento vistas en clases.  
  
3. Realice clasificación usando el algoritmo SVM considerando las variables independientes para estimar la variable dependiente (Status). Divida en training set y testing set. Obtenga las métricas de rendimiento vistas en clases.  
  
4. Realice Cross-Validation a los algoritmos implementados en los pasos 2 y 3. Obtenga una métrica global del proceso. Por último, revise si existe Overfitting dados los resultados del training y testing set.

Imagen que contiene Gráfico

Descripción generada automáticamenteA esto se refieren con Cross validation?

1. Seguirá trabajando con el dataset: Olivetti Faces. Como primer paso usted deberá cargar los datos y corroborar las dimensiones del dataset, el cual debe ser 400 x 4096  
     
   2. Sabiendo que el número de clusters (personas) es 40, aplique K-Means y cluster jerárquico. Mediante el método del codo, identifique cuántos clusters le entrega cada uno de los algoritmos. ¿Cuál le da mejor resultado?  
     
   3. Ahora aplique PCA al dataset. Quédese con un número de componentes principales tal que pueda capturar el 90% de la varianza explicada. ¿Cuántos componentes principales son?  
     
   4. Aplique K-Means y cluster jerárquico al dataset reducido por PCA. Mediante el método del codo, identifique cuántos clusters le entrega cada uno de los algoritmos. ¿Cuál algoritmo le da mejor resultado? Compare los resultados con y sin PCA. ¿Sirvió de algo aplicar PCA?  
     
   5. Divida el dataset en training y testing set. 80% training y 20% test. Use el coeficiente de Silhouette para estimar los valores de los parámetros de DBSCAN aplicado al training set.  
     
   6. Con los valores estimados, aplique DBSCAN al training set y luego haga predicciones al testing set. Compare que tan bien estimo las etiquetas el algoritmo con respecto a las verdaderas etiquetas del testing set.

Considerando lo aprendido en este módulo, implemente lo que Ud. crea necesario para mejorar los resultados del problema de decoding visto en clases