Notas Breves

IDM - 2020

Preprocesamiento de Datos

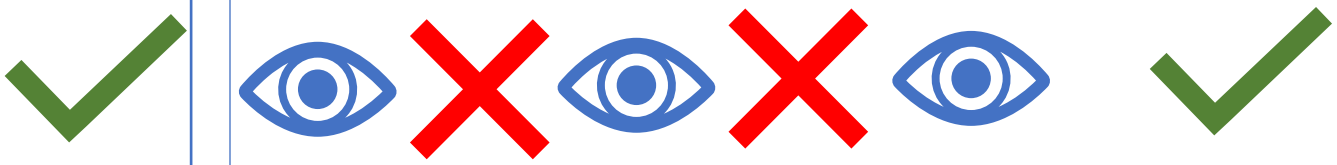
¿Por que necesitamos pre-procesar los datos?

- Los datos contenidos en una base de datos pueden estar incompletos, o presentar “ruido”:
  - Campos obsoletos o redundantes;
  - Valores faltantes;
  - Valores extremos;
  - Datos en formas no adecuadas para modelos de *data mining;*
  - Valores no consistentes con políticas, estándares o incluso el sentido común
- Generalmente el pre-procesamiento incluye:
  - Limpieza de Datos
  - Transformación de Datos
- El objetivo es minimizar el *GIGO*.

Limpieza de Datos

TABLE 2.1

Customer ID	Zip	Gender	Income	Age	Marital Status	Transaction Amount
1001	10048	M	75,000	C	M	5000
1002	J2S7K7	F	−40,000	40	W	4000
1003	90210		10,000,000	45	S	7000
1004	6269	M	50,000	0	S	1000
1005	55101	F	99,999	30	D	3000



¿Qué problemas observa en estos datos?

No parece respetar el formato americano de 5 dígitos…
En realidad, corresponde a la ciudad de St. Hyancinthe, Quebec, Canadá

Los códigos postales para *New England* comienzan con 0, pero ¿y si el campo es numérico en lugar de cadena?.
06269, corresponde a Storrs, Connecticut…

Valores Faltantes

- cars.txt (Barry Becker y Ron Kohavi)
  - 261 vehículos manufacturados entre 1970 y 1980.

	mpg	cubicinches	hp	brand
1	14.000	350	165	US
2	31.900		71	Europe
3	17.000	302	140	US
4	15.000	400	150	
5	37.700	89	62	Japan

Primeros 5 registros del archivo, con valores faltantes resaltados…

Una opción para tratar los valores faltantes es simplemente omitir los registros o campos con valores faltantes… Pero esto es peligroso. Puede existir un patrón sistemático en los valores faltantes, y omitirlos llevaría a un subconjunto sesgado. Schmueli, Patel y Bruce (2010) nos recuerdan que si solamente el 5 % de los valores faltan de un conjunto de 30 variables, y los valores faltantes se distribuyen uniformemente en el conjunto, el 80 % de los registros tendría por lo menos un valor faltante.

Valores Faltantes

- Algunos criterios usuales para reemplazar los valores faltantes son:
  - Reemplazar los faltantes por algún valor constante especificado por el analista.
  - Reemplazar los valores faltante con el promedio del campo (para variables numéricas) o por la moda (para variables categóricas).
  - Reemplazar los valores faltantes con un valor generado al azar dentro de la distribución observada de la variable.
  - Reemplazar los valores faltantes con valores *imputados* basados en otras características de los registros.

Valores Faltantes

	mpg	cubicinches	hp	brand
1	14.000	350	165	US
2	31.900	0	71	Europe
3	17.000	302	140	US
4	15.000	400	150	Missing
5	37.700	89	62	Japan

Reemplazar por constantes definidas por el usuario
0 para la variable *cubicinches*
Missing para la variable *brand*

	mpg	cubicinches	hp	brand
1	14.000	350	165	US
2	31.900	200.65	71	Europe
3	17.000	302	140	US
4	15.000	400	150	US
5	37.700	89	62	Japan

Reemplazar por la media o moda
- La variable *Brand* es categorica con moda US.
- La variable *cubicinches* es numérica con media 200.65

	mpg	cubicinches	hp	brand
1	14.000	350	165	US
2	31.900	450	71	Europe
3	17.000	302	140	US
4	15.000	400	150	Japan
5	37.700	89	62	Japan

Reemplazar por valores aleatorios dentro de la distribución de la variable
Si bien esto garantiza que las medidas de tendencia central y de dispersión se mantengan similares a las originales, es posible obtener casos sin sentido…

Valores Faltantes – Imputación

- Los métodos de imputación de datos faltantes hacen uso del conocimiento que el auto es Japones al calcular el valor de *cubicinches*.
- Nos preguntamos, ¿*Cuál es el valor mas probable para el valor faltante, conocidos los demás atributos para un registro en particular*?
- Es posible emplear para esto regresiones múltiples o CARTs…

Identificar Errores de Clasificación

- Controlemos la clasificación de las variables categóricas para asegurarnos que sea valida y consistente, por ejemplo para la variable *brand*:

TABLE 2.2

Brand	Frequency
USA	1
France	1
US	156
Europe	46
Japan	51

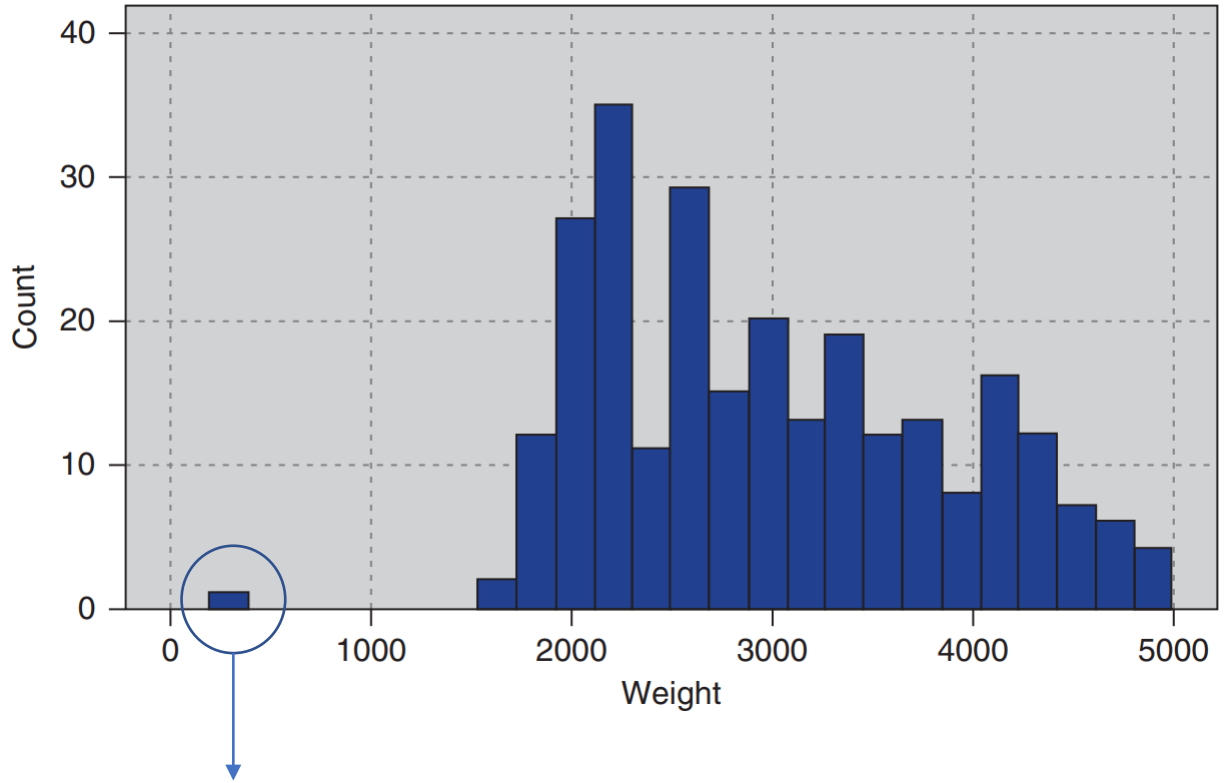
Errores de clasificación

Métodos Gráficos para Identificar *Outliers*

- Los *valores extremos* son valores contrarios a la tendencia de los datos. Es importante identificarlos dado que *pueden* representar errores en la captura de datos.

- Aun cuando los *valores extremos* sean valores **válidos** y no errores, varios métodos estadísticos son sensibles a la presencia de *outliers* y pueden entregar resultados incorrectos o inestables.

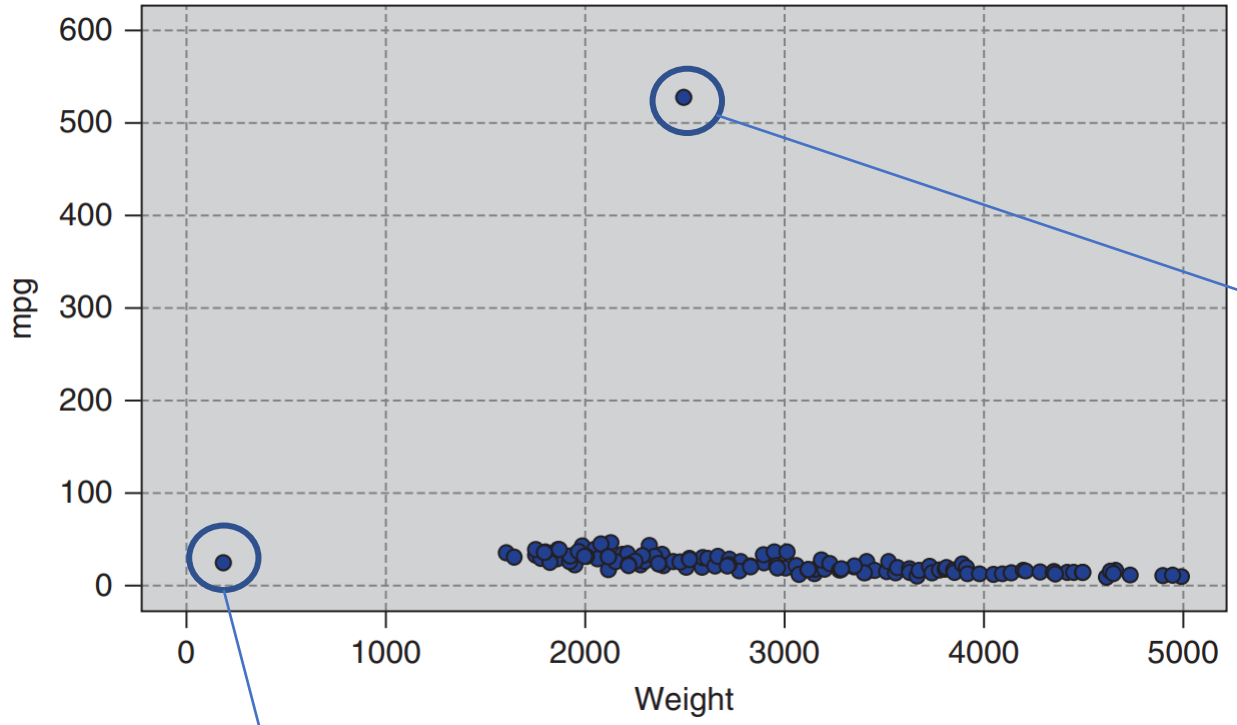
Métodos Gráficos para Identificar *Outliers*



Histograma de la variable peso del vehículo
¿Puede encontrar el *outlier*?

Aparece un vehículo con peso en el orden de los cientos en lugar de las miles de libras.
De hecho el menor valor de la variable es de 192.5 libras (87,3 kg)… quizás el peso correcto era de 1925 libras (873,2 kg)…

Métodos Gráficos para Identificar *Outliers*



Scatterplot bidimensional de la variable *mpg* vs. peso del vehículo

Vehículo de 500 millas por galón de combustible (213 km / litro)

Vehículo de 192.5 libras de peso

Un registro puede ser un *outlier* en una dimensión pero no en otra…

Transformación de datos

- Las variables tienden a presentar rangos que varían sensiblemente unos de otros.
- Algunos algoritmos de *data mining*, expuestos a diferencias en los rangos tienden a favorecer en el resultado a la variable con el mayor rango.
- Es por eso usual *normalizar* las variables numéricas para estandarizar la escala de los efectos que cada variable tiene en los resultados. Las redes neuronales se benefician de la normalización lo mismo que los algoritmos que hacen uso de medidas de distancias (*e.g. k-NNH)*

Sea X el valor original y X* el valor normalizado…

Normalización MIN-MAX

- Detecta cuan grande es el valor en relación al *min(X)* y escala la diferencia por el rango:

$$X^*_{mm} = \frac{X - \min(X)}{rango(X)} = \frac{X - \min(X)}{\max(X) - \min(X)}$$

Estadísticas descriptivas para la variable *weightlbs*

- Vehiculo ultraliviano de 1613 libras:

$$X^*_{mm} = \frac{X - \min(X)}{rango(X)} = \frac{1613 - 1613}{3384} = 0$$

- Un vehículo de rango medio tendrá un valor de 0,5
- El vehículo mas pesado tendrá un valor de 1.

Estandarización por puntajes Z

- Trabaja tomando la diferencia entre el valor del campo y el promedio del mismo y escala esta diferencia por la desviación estándar:

$$Z = \frac{X - mean(X)}{SD(X)}$$

- Para el vehículo mas liviano (1613 libras), tenemos: $Z = \frac{X - mean(X)}{SD(X)} = \frac{1613 - 3005{,}49}{852{,}49} \approx -1{,}63$
- Para un vehículo promedio (si existe), tenemos: $Z = \frac{X - mean(X)}{SD(X)} = \frac{3005{,}49 - 3005{,}49}{852{,}49} = 0$
- Para el vehículo mas pesado, tenemos: $Z = \frac{X - mean(X)}{SD(X)} = \frac{4997 - 3005{,}49}{852{,}49} \approx -2{,}34$

Escalado Decimal

- Asegura que cada valor normalizado se encuentre entre -1 y 1.

$$X^*_{decimal} = \frac{X}{10^d}$$

donde *d* es el numero de dígitos en el mayor valor absoluto.

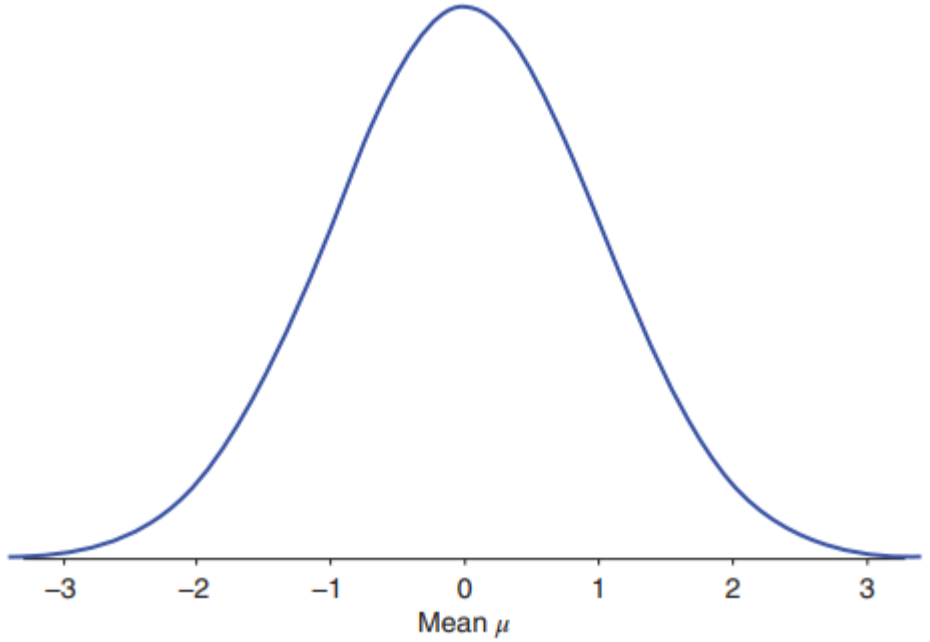
- En el caso del peso del vehículo, el mayor valor absoluto es |4997|=4997, que tiene *d* = 4 dígitos. Con esto el valor escalado para el peso mínimo y máximo, son

$$Min\colon X^*_{decimal} = \frac{1613}{10^4} = 0.1613$$

$$Max\colon X^*_{decimal} = \frac{4997}{10^4} = 0.4997$$

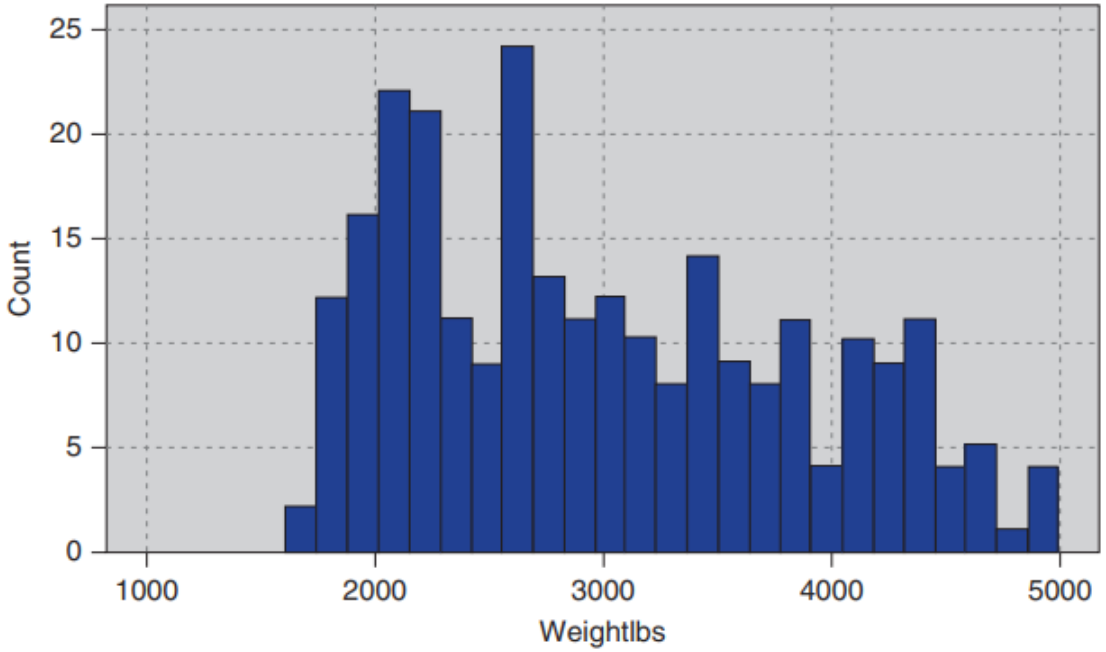
Transformaciones para obtener normalidad

- Algunos algoritmos de *data mining* y algunos métodos estadísticos requieren que las variables estén normalmente distribuidas.
- La estandarización z NO normaliza los datos (aunque es cierto que los mismos tendrán media 0 y desviación estándar 1 como la distribución normal estandarizada)

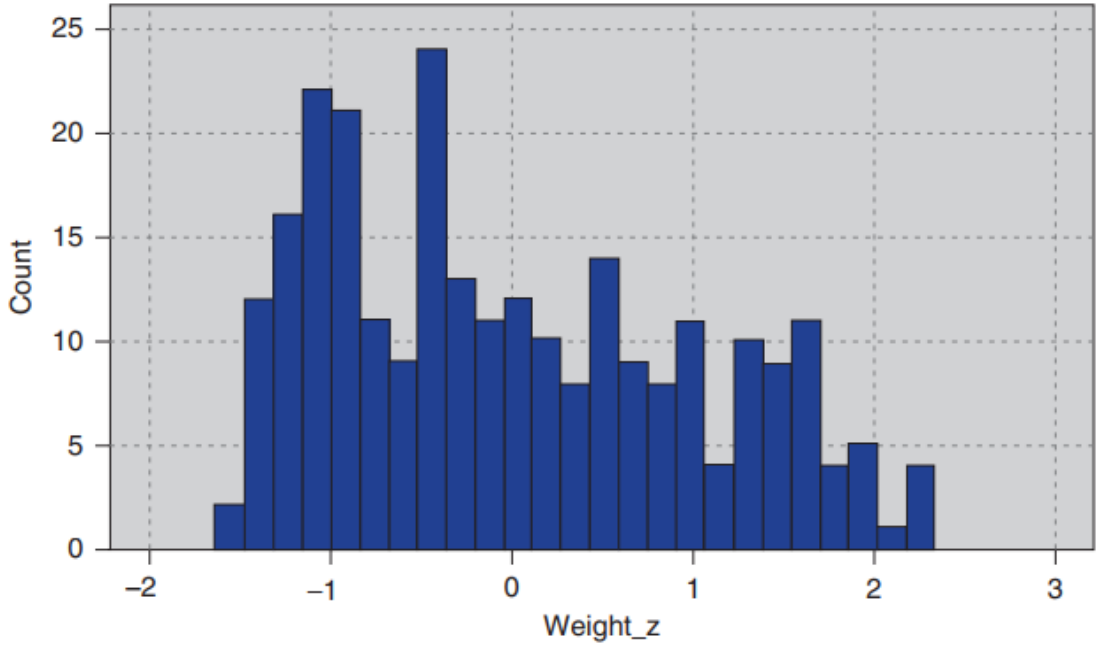


Transformaciones para obtener normalidad

Datos originales



Datos estandarizados por puntajes z



Sesgo derecho y falta de simetría (no están normalmente distribuidos)

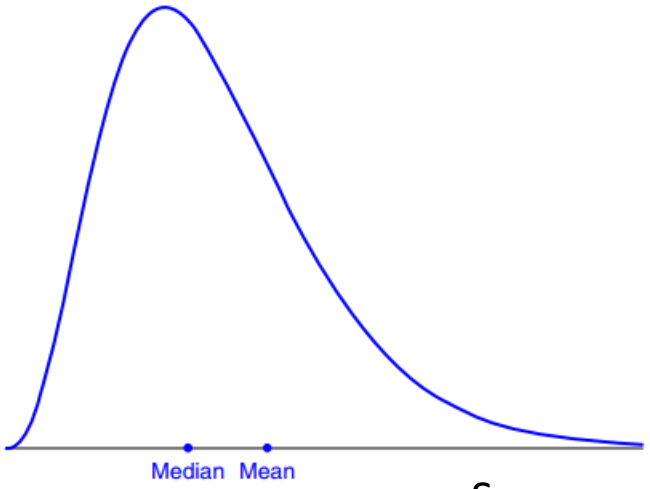
$$Sesgo = \frac{3(media - mediana)}{desviacion\ estandar}$$

Transformaciones para obtener normalidad

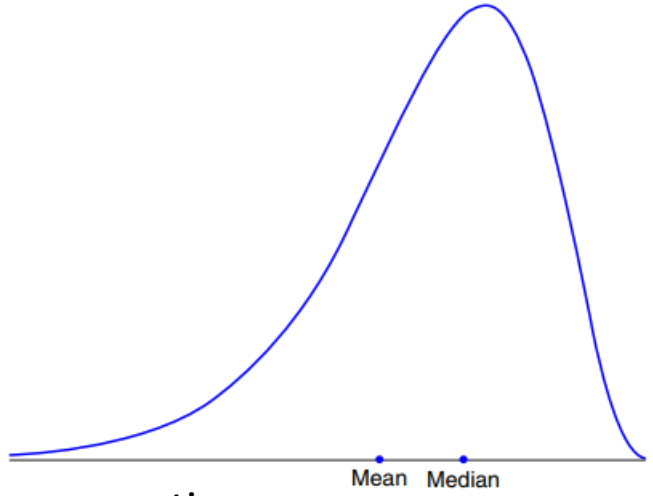
- Para la variable *weight* tenemos: $$Sesgo = \frac{3(media - mediana)}{desviacion\ estandar} = \frac{3(3005.490 - 2835)}{852{,}646} = 0.6$$

- Para la variable *weight_z* tenemos: $$Sesgo = \frac{3(media - mediana)}{desviacion\ estandar} = \frac{3(0 - (-0{,}2))}{1} = 0.6$$

La estandarización por puntaje z NO TIENE efecto en el SESGO.



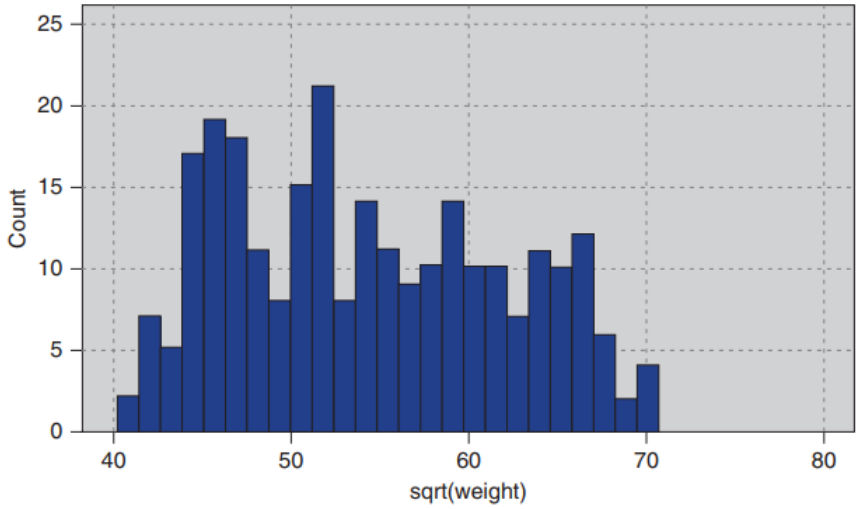
Sesgo positivo



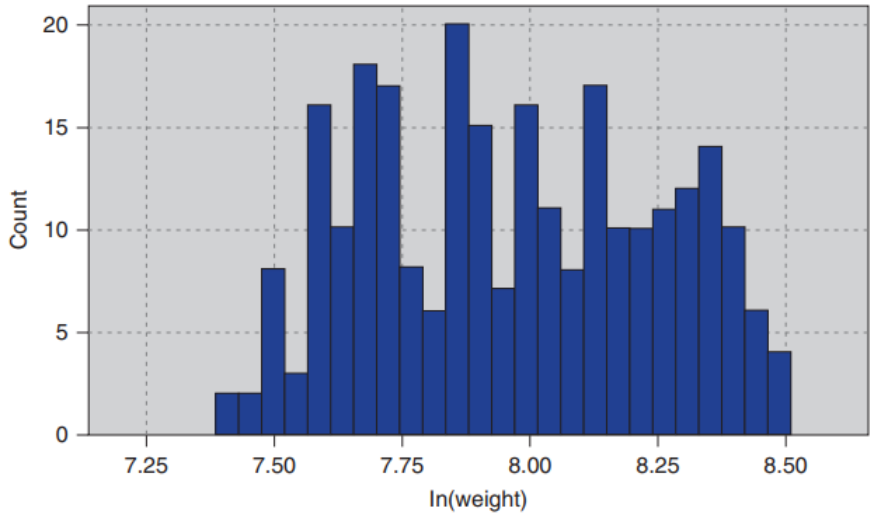
Sesgo negativo

Transformaciones para obtener normalidad

- Para eliminar el sesgo es necesario aplicar alguna **transformación** a los datos.
- Transformaciones comunes son:
  - Ln(*weight*)
  - √(*weight*)
  - 1/√(*weight*)

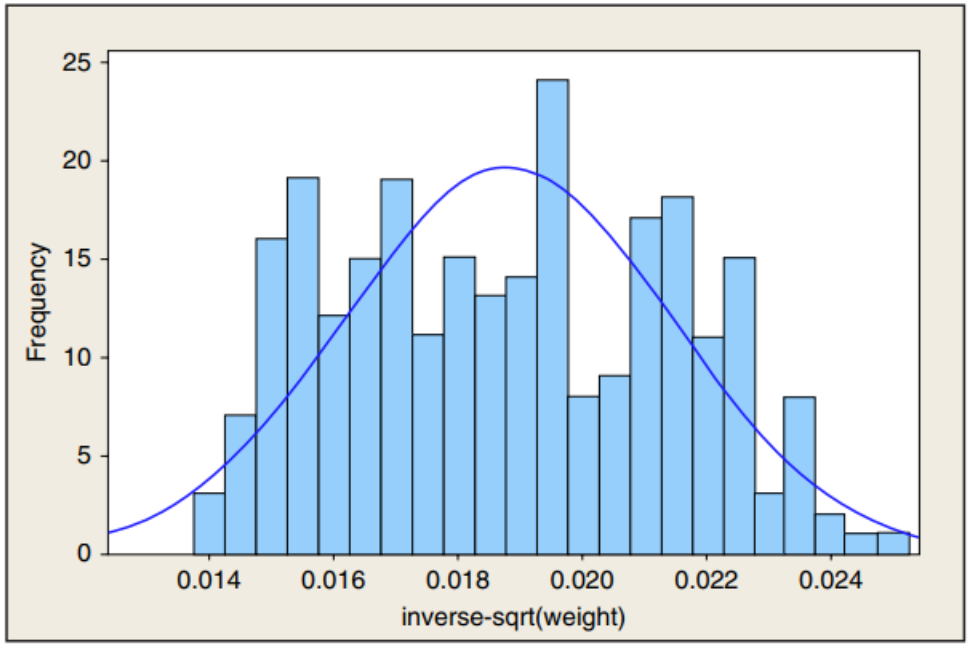


Sesgo ≈ 0,4



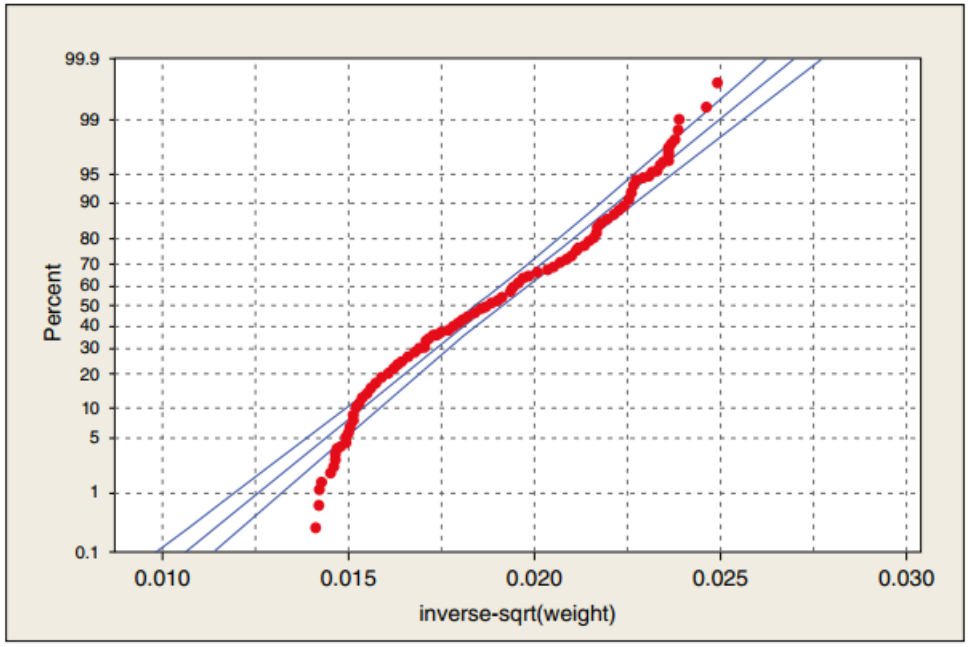
Sesgo ≈ 0,19

Transformaciones para obtener normalidad



Sesgo ≈ 0

- Si bien tenemos simetría aun no tenemos normalidad…
- Afortunadamente muchos algoritmos que requieren normalidad funcionan razonablemente si los datos son simétricos y unimodales.



Métodos numéricos para identificar *outliers*

- El método por puntaje z, indica que un valor es extremo (*outlier*) si su puntaje z es menor que -3 o mayor que 3.
- Cuidado con omitir automáticamente valores extremos de un conjunto.
- Recordar que tanto la media como la desviación estándar son susceptibles a valores extremos.
- **Métodos mas Robustos**:
  - Emplear el RIC = C3 – C1 (dispersión del 50 % central)
  - Un valor es extremo si:
    a) Está ubicado a 1.5(RIC) o mas por debajo de C1, o
    b) Está ubicado a 1.5(RIC) o mas por encima de C3
  - Ej. Si el 25 % (C1 = 70) y 75 % (C2 = 80), la mitad de los valores caen entre 70 y 80. Por lo tanto RIC = 80 – 70 = 10. Un valor se identificaría como extremo si:
    a) es mas pequeño que C1 – 1.5(RIC) = 70 – 1.5(10) = 55, o
    b) es mas grande que C3 + 1.5(RIC) = 80 + 1,5(10) = 95

Variables Indicadoras

- Algunos métodos (como la regresión) requieren que los predictores sean numéricos. Si pretendemos emplear predictores categóricos en ese caso es necesario **recodificar** la variable en una o mas *variables indicadoras (dummies)*.

- Se trata de variables que toman solamente los valores **0** o **1**.
  - Ej. Predictor categórico *sex (hombre, mujer)* puede ser recodificado en la variable *sex_flag*:

  If *sex = female* then *sex_flag* = 0; if *sex = male* then *sex_flag* = 1

- Si el predictor categórico toma *k ≥ 3* valores posibles, se definen *k-1* variables dummies, y se emplea la categoría no asignada como *categoría de referencia.*
  - Ej. Predictor categórico *región {north, east, south, west}*, *k = 4*, por lo que definimos solamente *k-1 = 3* varibales

  North_flag: If *región = north* then *north_flag* = 1 else *north_flag = 0;*

  East_flag: If *región = east* then *east_flag* = 1 else *east_flag = 0;*

  South_flag: If *región =* south then *south_flag = 1* else *south_flag = 0*;

  no se necesita la variable indicadora para *región = west*, ya que esta unívocamente determinada por los ceros de las otras variables indicadoras. La categoría no asignada se transforma en la categoría de referencia, es decir, el valor de *north_flag* es la *región = north* **comparada** con *región = west.*

Transformando variables categóricas en variables numéricas

- ¿Qué pasaría si hiciéramos?

Region	*Region_num*
North	1
East	2
South	3
West	4

Es un error común…
Los algoritmos pensarían entonces que las regiones están ordenadas
West > South > East > North
West es tres veces mas cercano a South comparado con North… etc.

- ¿Y en este caso?

Survey response	*Survey Response_num*
Always	4
Usually	3
Sometimes	2
Never	1

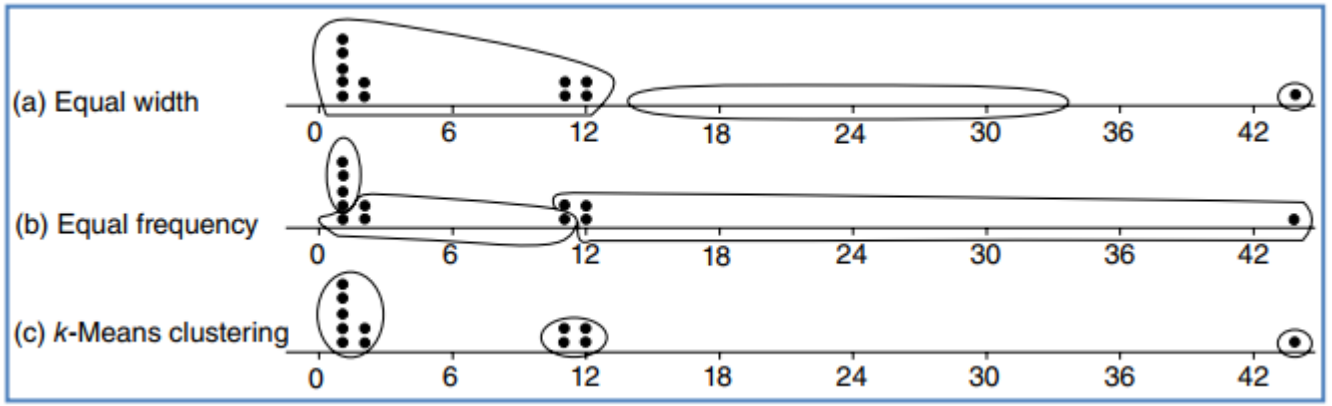
¿Debiéramos haber elegido *never = 0* en lugar de 1?
¿Está *always* mas próxima a *usually* que *usually* a *sometimes*?

Categorización de variables numéricas

- Algunos algoritmos prefieren variables categóricas en lugar de predictores continuos.
- En este caso estamos obligados a construir *bins* o bandas:
    1. *Bandas de Igual Ancho*: divide el predictor numérico en *k* categorías de igual amplitud, donde *k* es elegido por el analista.
    2. *Bandas de Igual Frecuencia:* divide el predictor numérico en *k* categorías, en cada una de las cuales hay *k/n* registros, donde *n* es la cantidad total de registros.
    3. *Bandas por conglomeración*: emplea un algoritmo de clustering (e.g. k-means) para calcular automáticamente el particionamiento “optimo”.
    4. *Bandas basadas en el valor predictivo:* los métodos 1 a 3 ignoran la variable objetivo; en este caso, se divide el predictor numérico basándose en el efecto que cada partición tiene en la variable objetiva.

Categorización de variables numéricas

- Sea el conjunto X = {1,1,1,1,1,2,2,11,11,12,12,44} a discretizar en *k* = *3* categorías.



Reclasificación de variables categóricas

- Es el equivalente de categorizar variables numéricas.
- Útil cuando una variable categórica presenta demasiados valores diferentes.
- La regresión logística o el algoritmo C4.5 tienen un desempeño sub-optimo cuando encuentran predictores con demasiados valores.

Remoción de variables

- Es útil remover variables que no ayuden en el análisis en curso, sin que importe la tarea o algoritmo de *data mining*.
  - Variables unarias
  - Variables que son casi completamente unarias
- Una *variable unaria* toma un único valor, por lo que en esencia una variable de este tipo es equivalente a una constante. (Ej. Campo sexo en una muestra de estudiantes de una escuela de mujeres).
- Es común (aunque cuestionable) remover del análisis:
  - Variables para las que 90 % o mas de los valores son *missing*.
  - Variables fuertemente correlacionadas.

Remoción de Registros Duplicados

- La existencia de registros duplicados lleva a una sobre-ponderacion de los valores de esos registros, por lo que si realmente están duplicados se deben remover las replicas.