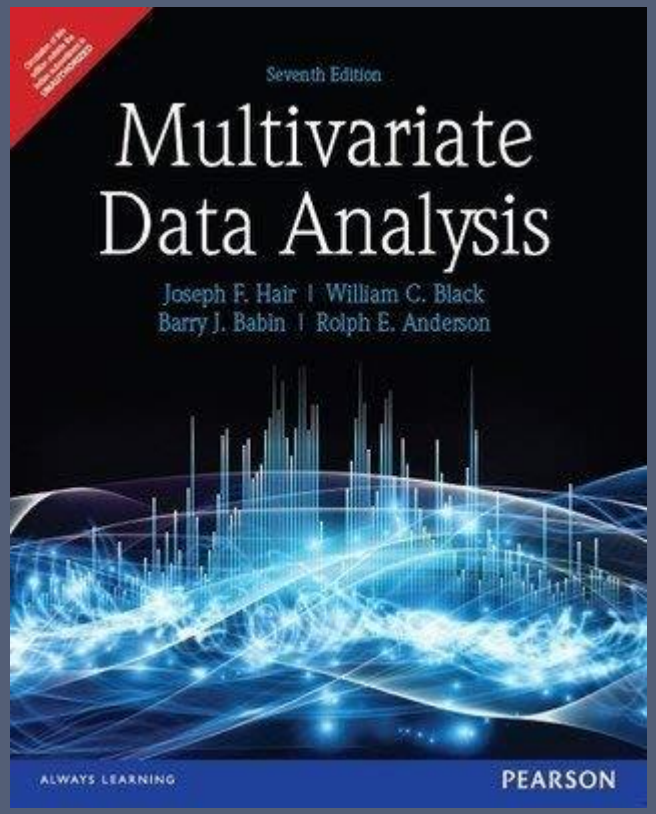
Index Class

1. Análisis del Problema
2. Análisis Univariable y Multivariable
3. Limpieza de los datos
4. Normalización

Index Class



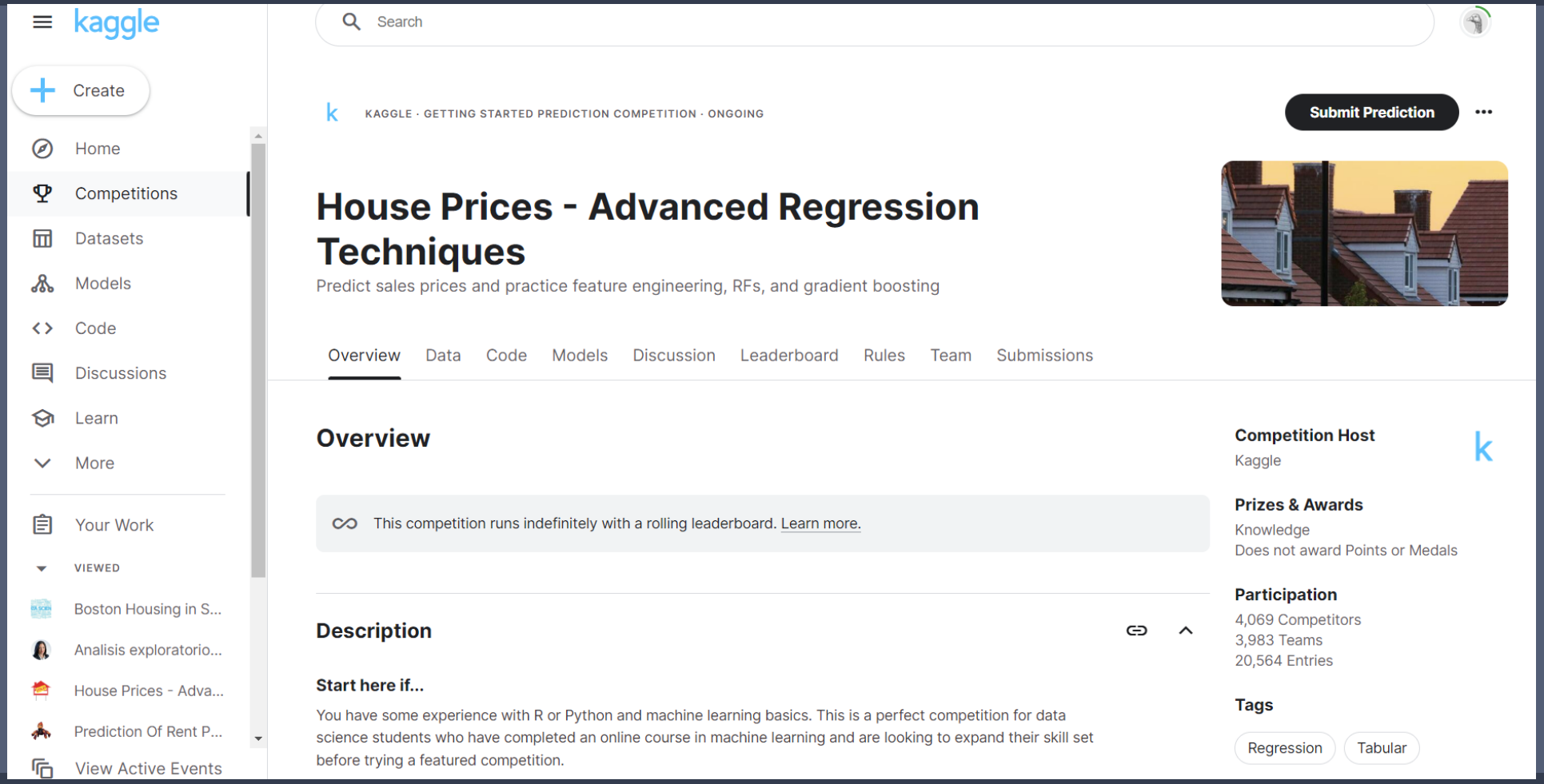
Hair et al., 2013, Multivariate Data Analysis, 7th Edition

1. Análisis del Problema

Siempre, partimos de entender el problema y los datos que tenemos.

- La relevancia de la variable pronóstico respecto de las otras variables.
- La importancia de la variable.
- Solapamiento con otras variables.

1. Análisis del Problema



1. Análisis del Problema

Evaluation

Goal

It is your job to predict the sales price for each house. For each Id in the test set, you must predict the value of the SalePrice variable.

Metric

Submissions are evaluated on Root-Mean-Squared-Error (RMSE) between the logarithm of the predicted value and the logarithm of the observed sales price. (Taking logs means that errors in predicting expensive houses and cheap houses will affect the result equally.)

1. Análisis del Problema

1. **MSSubClass**: clase de construcción
2. **MSZoning**: clasificación de la zona
3. **LotFrontage**: pies lineales de calle de la parcela
4. **LotArea**: tamaño de la parcela en pies cuadrados
5. **Street**: tipo de acceso por carretera
6. **Alley**: tipo de acceso al callejón
7. **LotShape**: forma de la parcela
8. **LandContour**: planitud de la parcela
9. **Utilities**: servicios públicos disponibles
10. **LotConfig**: Configuración de parcela
11. **LandSlope**: pendiente de la parcela
12. **Neighborhood**: ubicación física dentro de los límites de la ciudad de Ames
13. **Condition1**: proximidad a la carretera principal o al ferrocarril
14. **Condition2**: proximidad a la carretera principal o al ferrocarril (si hay un segundo)
15. **BldgType**: tipo de vivienda
16. **HouseStyle**: estilo de vivienda
17. **OverallQual**: calidad general del material y del acabado
18. **OverallCond**: condición general
19. **YearBuilt**: fecha original de construcción
20. **YearRemodAdd**: fecha de remodelación
21. **RoofStyle**: tipo de cubierta
22. **RoofMatl**: material del techo
23. **Exterior1st**: revestimiento exterior de la casa
24. **Exterior2nd**: revestimiento exterior de la casa (si hay más de un material)
25. **MasVnrType**: tipo de revestimiento de mampostería
26. **MasVnrArea**: área de revestimiento de mampostería en pies cuadrados
27. **ExterQual**: calidad del material exterior
28. **ExterCond**: estado del material en el exterior
29. **Foundation**: tipo de cimentación
30. **BsmtQual**: altura del sótano
31. **BsmtCond**: estado general del sótano
32. **BsmtExposure**: paredes del sótano a nivel de calle o de jardín
33. **BsmtFinType1**: calidad del área acabada del sótano
34. **BsmtFinSF1**: pies cuadrados de la superficie acabada tipo 1
35. **BsmtFinType2**: calidad de la segunda superficie acabada (si existe)
36. **BsmtFinSF2**: Pies cuadrados de la superficie acabada tipo 2
37. **BsmtUnfSF**: pies cuadrados del área sin terminar del sótano
38. **TotalBsmtSF**: pies cuadrados totales del sótano
39. **Heating**: tipo de calefacción
40. **HeatingQC**: calidad y estado de la calefacción
41. **CentralAir**: aire acondicionado central
42. **Electrical**: sistema eléctrico
43. **1erFlrSF**: área en pies cuadrados de la primera planta (o planta baja)
44. **2ndFlrSF**: área en pies cuadrados de la segunda planta
45. **LowQualFinSF**: pies cuadrados acabados de baja calidad (todos los pisos)
46. **GrLivArea**: superficie habitable por encima del nivel del suelo en pies cuadrados
47. **BsmtFullBath**: cuartos de baño completos en el sótano
48. **BsmtHalfBath**: medio baño del sótano
49. **FullBath**: baños completos sobre el nivel del suelo
50. **HalfBath**: medios baños sobre el nivel del suelo
51. **Bedroom**: número de dormitorios por encima del nivel del sótano
52. **Kitchen**: número de cocinas
53. **KitchenQual**: calidad de la cocina
54. **TotRmsAbvGrd**: total de habitaciones por encima del nivel del suelo (no incluye baños)
55. **Functional**: valoración de la funcionalidad de la vivienda
56. **Fireplaces**: número de chimeneas
57. **FireplaceQu**: calidad de la chimenea
58. **GarageType**: ubicación del garaje
59. **GarageYrBlt**: año de construcción del garaje
60. **GarageFinish**: acabado interior del garaje
61. **GarageCars**: tamaño del garaje en capacidad de coches
62. **GarageArea**: tamaño del garaje en pies cuadrados
63. **GarageQual**: calidad de garaje
64. **GarageCond**: condición de garaje
65. **PavedDrive**: calzada asfaltada
66. **WoodDeckSF**: area de plataforma de madera en pies cuadrados
67. **OpenPorchSF**: área de porche abierto en pies cuadrados
68. **EnclosedPorch**: área de porche cerrada en pies cuadrados
69. **3SsnPorch**: área de porche de tres estaciones en pies cuadrados
70. **ScreenPorch**: superficie acristalada del porche en pies cuadrados
71. **PoolArea**: área de la piscina en pies cuadrados
72. **PoolQC**: calidad de la piscina
73. **Fence**: calidad de la valla
74. **MiscFeature**: característica miscelánea no cubierta en otras categorías
75. **MiscVal**: valor en dólares de la característica miscelánea
76. **MoSold**: mes de venta
77. **YrSold**: año de venta
78. **SaleType**: tipo de venta
79. **SaleCondition**: Condiciones de venta

1. Análisis del Problema

1. **MSSubClass**: clase de construcción
2. **MSZoning**: clasificación de la zona
3. **LotFrontage**: pies lineales de calle de la parcela
4. **LotArea**: tamaño de la parcela en pies cuadrados
5. **Street**: tipo de acceso por carretera
6. **Alley**: tipo de acceso al callejón
7. **LotShape**: forma de la parcela
8. **LandContour**: planitud de la parcela
9. **Utilities**: servicios públicos disponibles
10. **LotConfig**: Configuración de parcela
11. **LandSlope**: pendiente de la parcela
12. **Neighborhood**: ubicación física dentro de los límites de la ciudad de Ames
13. **Condition1**: proximidad a la carretera principal o al ferrocarril
14. **Condition2**: proximidad a la carretera principal o al ferrocarril (si hay un segundo)
15. **BldgType**: tipo de vivienda
16. **HouseStyle**: estilo de vivienda
17. **OverallQual**: calidad general del material y del acabado
18. **OverallCond**: condición general
19. **YearBuilt**: fecha original de construcción
20. **YearRemodAdd**: fecha de remodelación
21. **RoofStyle**: tipo de cubierta
22. **RoofMatl**: material del techo
23. **Exterior1st**: revestimiento exterior de la casa
24. **Exterior2nd**: revestimiento exterior de la casa (si hay más de un material)
25. **MasVnrType**: tipo de revestimiento de mampostería
26. **MasVnrArea**: área de revestimiento de mampostería en pies cuadrados
27. **ExterQual**: calidad del material exterior
28. **ExterCond**: estado del material en el exterior
29. **Foundation**: tipo de cimentación
30. **BsmtQual**: altura del sótano
31. **BsmtCond**: estado general del sótano
32. **BsmtExposure**: paredes del sótano a nivel de calle o de jardín
33. **BsmtFinType1**: calidad del área acabada del sótano
34. **BsmtFinSF1**: pies cuadrados de la superficie acabada tipo 1
35. **BsmtFinType2**: calidad de la segunda superficie acabada (si existe)
36. **BsmtFinSF2**: Pies cuadrados de la superficie acabada tipo 2
37. **BsmtUnfSF**: pies cuadrados del área sin terminar del sótano
38. **TotalBsmtSF**: pies cuadrados totales del sótano
39. **Heating**: tipo de calefacción
40. **HeatingQC**: calidad y estado de la calefacción
41. **CentralAir**: aire acondicionado central
42. **Electrical**: sistema eléctrico
43. **1erFlrSF**: área en pies cuadrados de la primera planta (o planta baja)
44. **2ndFlrSF**: área en pies cuadrados de la segunda planta
45. **LowQualFinSF**: pies cuadrados acabados de baja calidad (todos los pisos)
46. **GrLivArea**: superficie habitable por encima del nivel del suelo en pies cuadrados
47. **BsmtFullBath**: cuartos de baño completos en el sótano
48. **BsmtHalfBath**: medio baño del sótano
49. **FullBath**: baños completos sobre el nivel del suelo
50. **HalfBath**: medios baños sobre el nivel del suelo
51. **Bedroom**: número de dormitorios por encima del nivel del sótano
52. **Kitchen**: número de cocinas
53. **KitchenQual**: calidad de la cocina
54. **TotRmsAbvGrd**: total de habitaciones por encima del nivel del suelo (no incluye baños)
55. **Functional**: valoración de la funcionalidad de la vivienda
56. **Fireplaces**: número de chimeneas
57. **FireplaceQu**: calidad de la chimenea
58. **GarageType**: ubicación del garaje
59. **GarageYrBlt**: año de construcción del garaje
60. **GarageFinish**: acabado interior del garaje
61. **GarageCars**: tamaño del garaje en capacidad de coches
62. **GarageArea**: tamaño del garaje en pies cuadrados
63. **GarageQual**: calidad de garaje
64. **GarageCond**: condición de garaje
65. **PavedDrive**: calzada asfaltada
66. **WoodDeckSF**: area de plataforma de madera en pies cuadrados
67. **OpenPorchSF**: área de porche abierto en pies cuadrados
68. **EnclosedPorch**: área de porche cerrada en pies cuadrados
69. **3SsnPorch**: área de porche de tres estaciones en pies cuadrados
70. **ScreenPorch**: superficie acristalada del porche en pies cuadrados
71. **PoolArea**: área de la piscina en pies cuadrados
72. **PoolQC**: calidad de la piscina
73. **Fence**: calidad de la valla
74. **MiscFeature**: característica miscelánea no cubierta en otras categorías
75. **MiscVal**: valor en dólares de la característica miscelánea
76. **MoSold**: mes de venta
77. **YrSold**: año de venta
78. **SaleType**: tipo de venta
79. **SaleCondition**: Condiciones de venta

2. Análisis Univariable y Multivariable

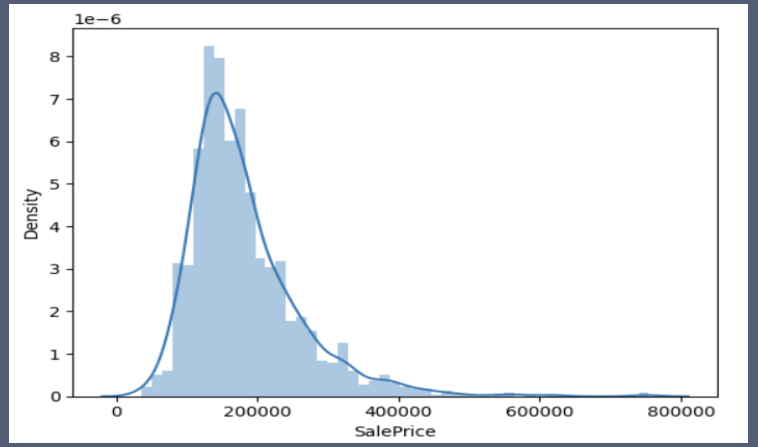
Estudiamos la variable Objetivo “SalePrice”.

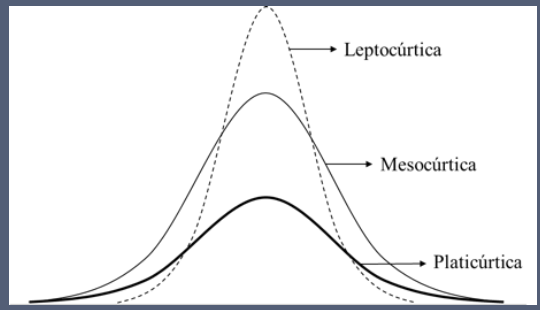
- Una desviación con respecto a la distribución normal.
- Una asimetría positiva (asimétrica. No centrada). Precios altos poco frecuentes y variados
- Leptocúrtica.

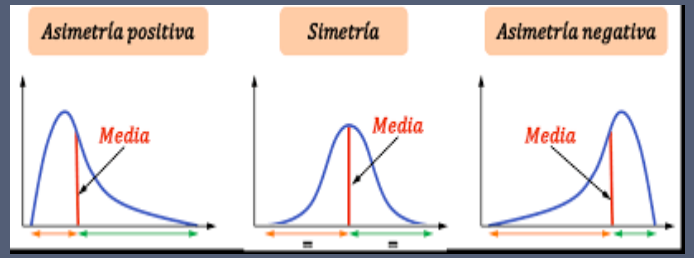
```
X['SalePrice'].describe()
```

```
count      1460.000000
mean     180921.195890
std       79442.502883
min       34900.000000
25%      129975.000000
50%      163000.000000
75%      214000.000000
max      755000.000000
Name: SalePrice, dtype: float64
```

```
X['SalePrice'].describe() #no tiene valores negativos! genial
sns.distplot(X['SalePrice'])
```



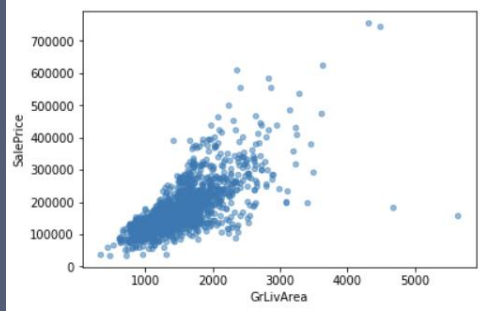




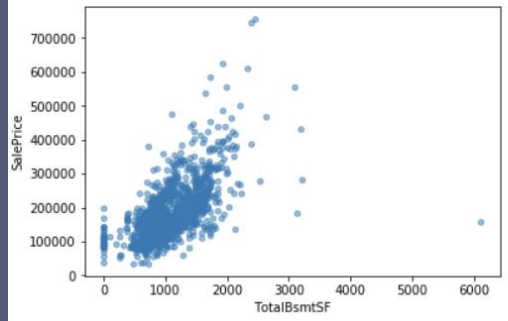

```
# Asimetría y curtosis:
("Skewness: %f" % df_train['SalePrice'].skew())
("Kurtosis: %f" % df_train['SalePrice'].kurt())
```

2. Análisis Univariable y Multivariable

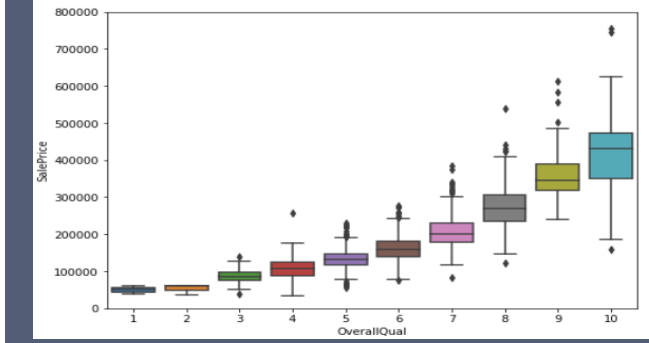
```
# Diagrama de dispersión grlivarea/saleprice:

var = 'GrLivArea'
data = pd.concat([df_train['SalePrice'], df_train[var]], axis=1)
data.plot.scatter(x=var, y='SalePrice', alpha = 0.5);
```




```
# Diagrama de dispersión totalbsmtsf/saleprice:

var = 'TotalBsmtSF'
data = pd.concat([df_train['SalePrice'], df_train[var]], axis=1)
data.plot.scatter(x=var, y='SalePrice', alpha = 0.5);
```



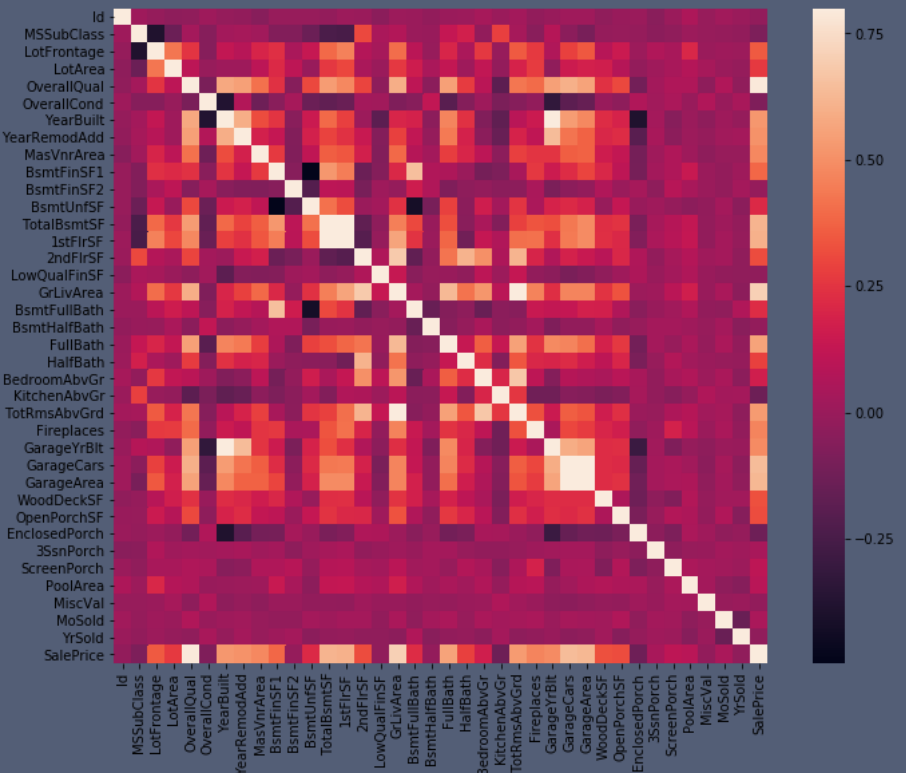

```
# Diagrama de cajas overallqual/saleprice:

var = 'OverallQual'
data = pd.concat([df_train['SalePrice'], df_train[var]], axis=1)
f, ax = plt.subplots(figsize=(8, 6))
fig = sns.boxplot(x=var, y="SalePrice", data=data)
fig.axis(ymin=0, ymax=800000);
```

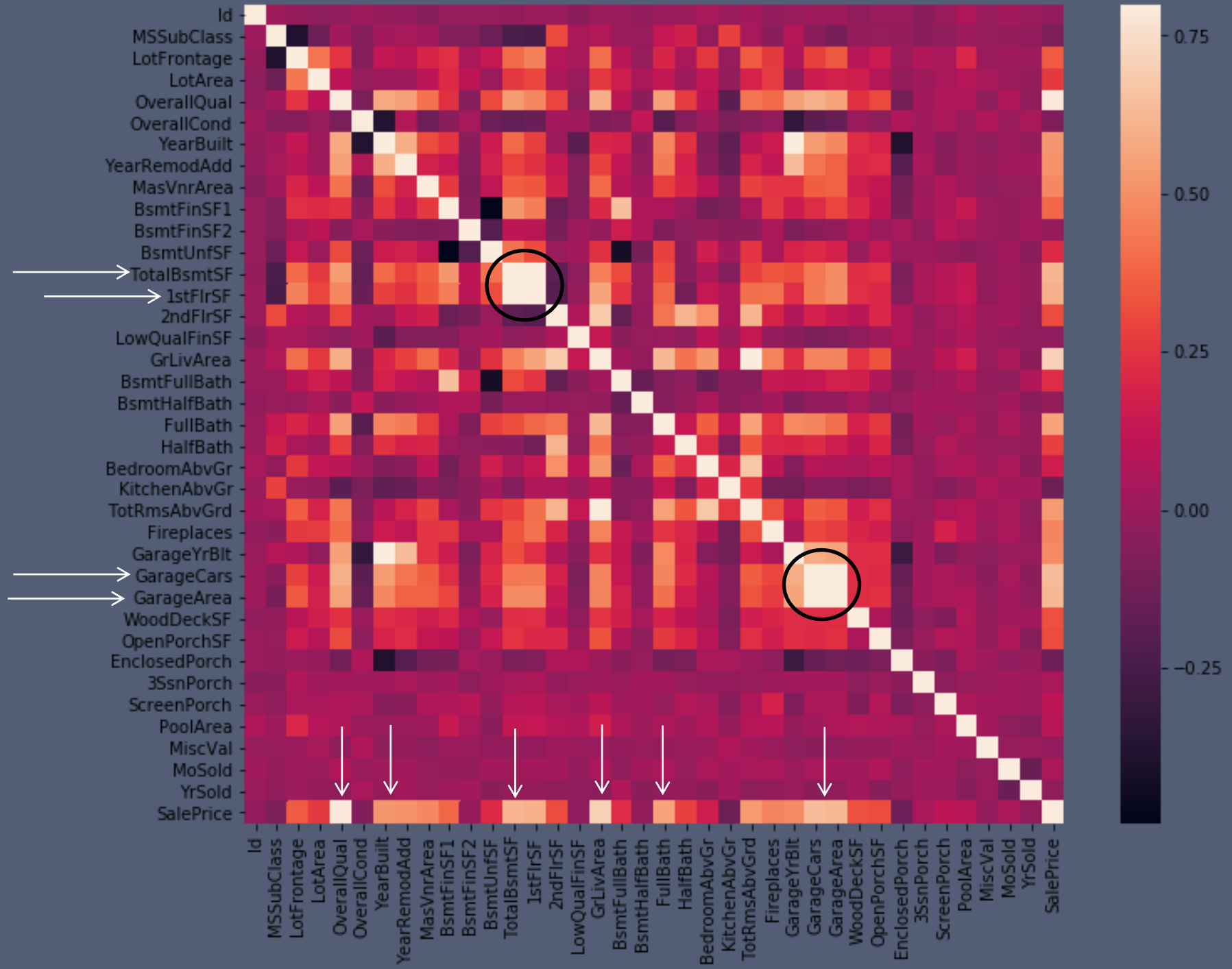


- Primeros análisis exploratorios:
- 'GrLivArea' y 'TotalBsmtSF' tienen una relación lineal positiva con 'SalePrice.
- En el caso de 'TotalBsmtSF', la pendiente de esta relación es mucho más alta.
- 'OverallQual' también parece estar relacionadas con 'SalePrice' . Se puede ver en el boxplot.

2. Análisis Univariable y Multivariable

Analizamos las correlaciones:





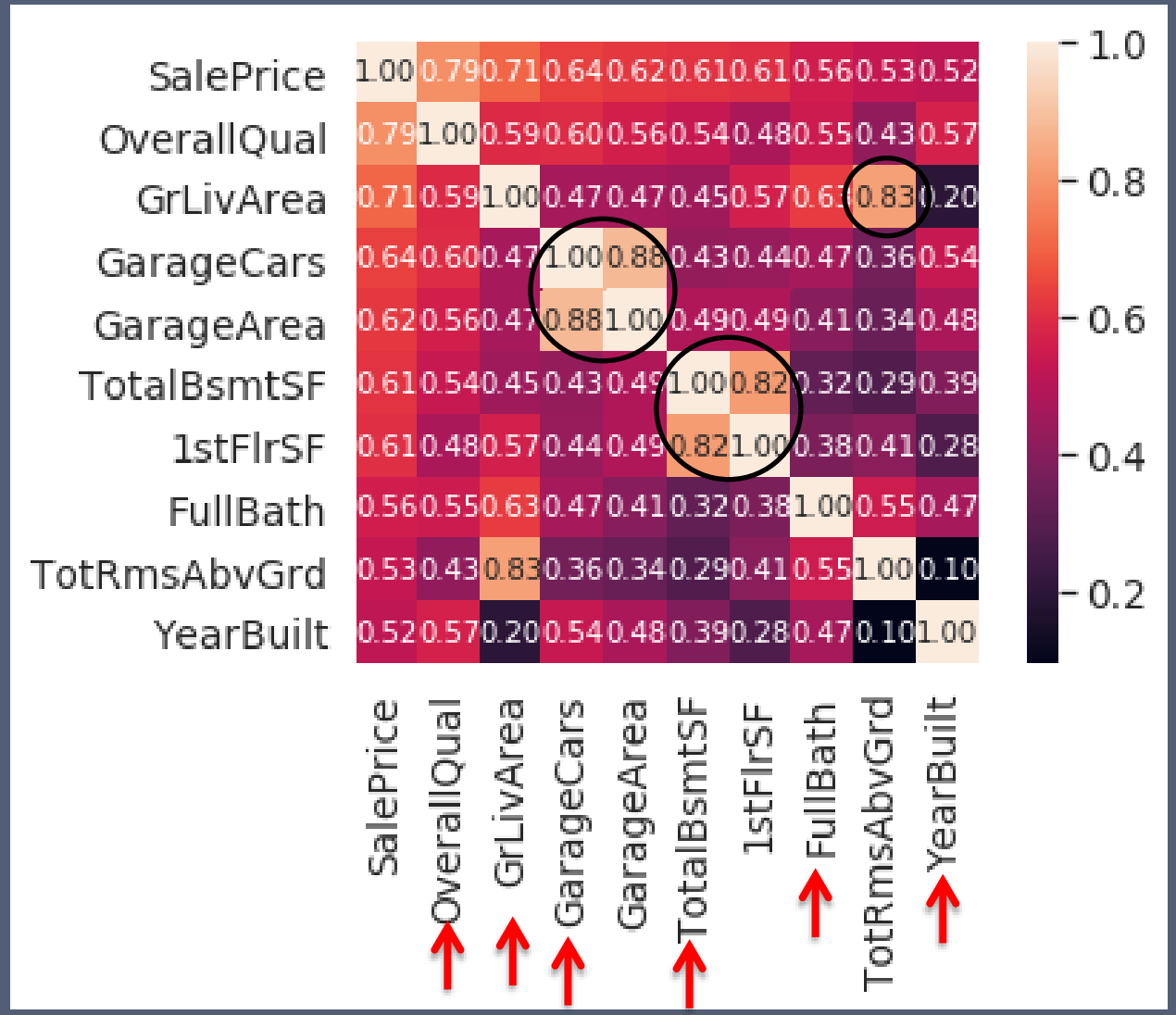
Variables que más llaman la atención en el heat map.

Parece haber una correlación muy fuerte que indica multicolinealidad (dan la misma info y podemos quitar una de ellas del análisis).

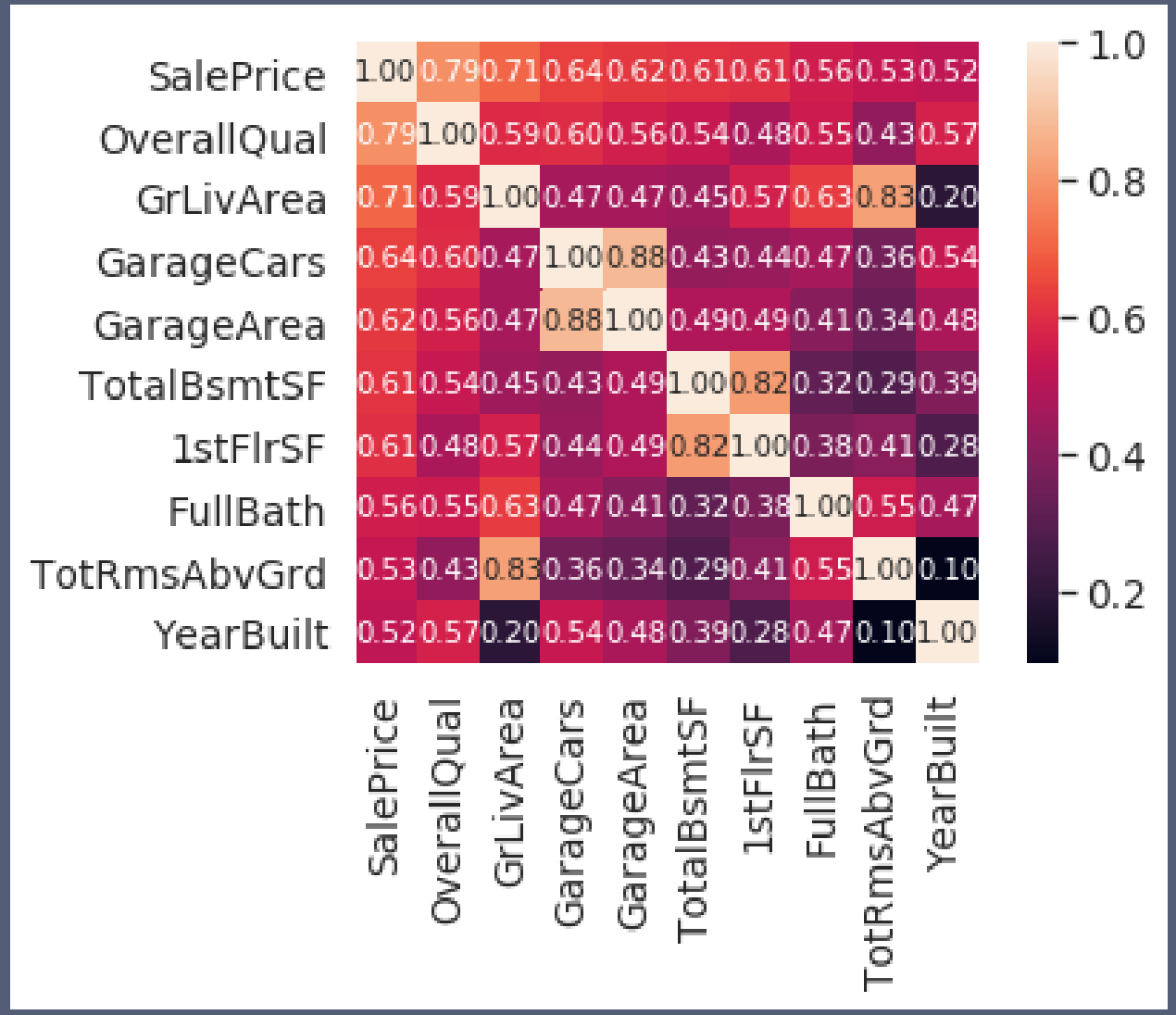
Con respecto a las correlaciones de la variable 'SalePrice', destacan:

('GrLivArea', 'TotalBsmtSF' y 'OverallQual'), pero hay otras que también deberían ser tenidas en cuenta.

```python
#Ampliación matriz, para variable que se selecciona muestra las k más correlacioandas
k = 10 #numero de variables
cols = corrmat.nlargest(k, 'TotalBsmtSF')['TotalBsmtSF'].index
cm = np.corrcoef(X[cols].values.T)
sns.set(font_scale=1.25)
hm = sns.heatmap(cm, cbar=True, annot=True, square=True, fmt='.2f', annot_kws={'size': 10}, yticklabels=cols.values, xticklabels=cols.values)
plt.show()
```

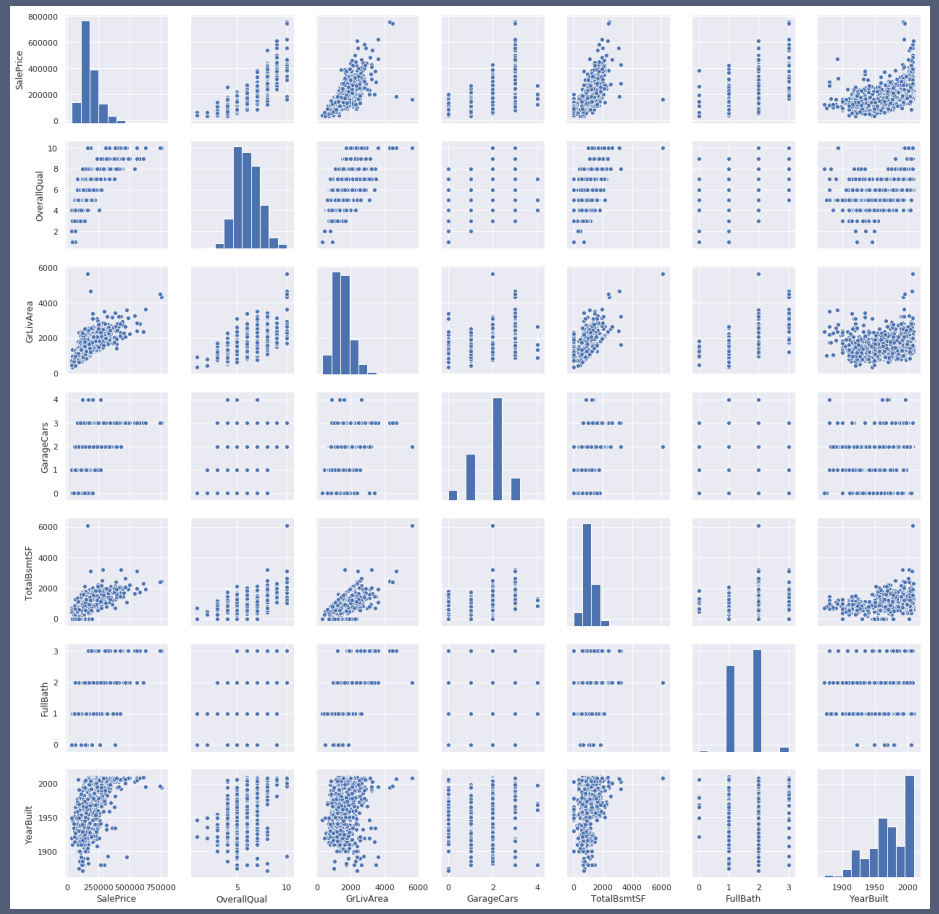


```python
#Ampliación matriz, para variable que se selecciona muestra las k más correlacioandas
k = 10 #numero de variables
cols = corrmat.nlargest(k, 'TotalBsmtSF')['TotalBsmtSF'].index
cm = np.corrcoef(X[cols].values.T)
sns.set(font_scale=1.25)
hm = sns.heatmap(cm, cbar=True, annot=True, square=True, fmt='.2f', annot_kws={'size': 10}, yticklabels=cols.values, xticklabels=cols.values)
plt.show()
```



2. Análisis Univariable y Multivariable

```
# Scatter plot:

sns.set()
cols = ['SalePrice', 'OverallQual', 'GrLivArea', 'GarageCars', 'TotalBsmtSF', 'FullBath', 'YearBuilt']
sns.pairplot(df_train[cols], size = 2.5)
plt.show();
```




SalePrice
OverallQual
GrLivArea
GarageCars
TotalBsmtSF
FullBath
YearBuilt
SalePrice
OverallQual
GrLivArea
GarageCars
TotalBsmtSF
FullBath
YearBuilt

3. Limpieza de Datos

	Total	Percent
PoolQC	1453	0.995205
MiscFeature	1406	0.963014
Alley	1369	0.937671
Fence	1179	0.807534
FireplaceQu	690	0.472603
LotFrontage	259	0.177397
GarageCond	81	0.055479
GarageType	81	0.055479
GarageYrBlt	81	0.055479
GarageFinish	81	0.055479
GarageQual	81	0.055479
BsmtExposure	38	0.026027
BsmtFinType2	38	0.026027
BsmtFinType1	37	0.025342
BsmtCond	37	0.025342
BsmtQual	37	0.025342
MasVnrArea	8	0.005479
MasVnrType	8	0.005479
Electrical	1	0.000685
Utilities	0	0.000000

```
# Missing data:

total = df_train.isnull().sum().sort_values(ascending = False)
percent = (df_train.isnull().sum() / df_train.isnull().count()).sort_values(ascending = False)
missing_data = pd.concat([total, percent], axis = 1, keys = ['Total', 'Percent'])
missing_data.head(20)
```

Miramos:
Variables con altos nulos
Variables con igual % de nulos,
Correlaciones entre variables
Correlaciones con la Y de pronóstico

3. Limpieza de Datos

	Total	Percent
PoolQC	1453	0.995205
MiscFeature	1406	0.963014
Alley	1369	0.937671
Fence	1179	0.807534
FireplaceQu	690	0.472603
LotFrontage	259	0.177397
GarageCond	81	0.055479
GarageType	81	0.055479
GarageYrBlt	81	0.055479
GarageFinish	81	0.055479
GarageQual	81	0.055479
BsmtExposure	38	0.026027
BsmtFinType2	38	0.026027
BsmtFinType1	37	0.025342
BsmtCond	37	0.025342
BsmtQual	37	0.025342
MasVnrArea	8	0.005479
MasVnrType	8	0.005479
Electrical	1	0.000685
Utilities	0	0.000000

```
# Missing data:

total = df_train.isnull().sum().sort_values(ascending = False)
percent = (df_train.isnull().sum() / df_train.isnull().count()).sort_values(ascending = False)
missing_data = pd.concat([total, percent], axis = 1, keys = ['Total', 'Percent'])
missing_data.head(20)
```

RULES OF THUMB 1

How Much Missing Data Is Too Much?

- Missing data under 10 percent for an individual case or observation can generally be ignored, except when the missing data occurs in a specific nonrandom fashion (e.g., concentration in a specific set of questions, attrition at the end of the questionnaire, etc.) [19, 20]
- The number of cases with no missing data must be sufficient for the selected analysis technique if replacement values will not be substituted (imputed) for the missing data

RULES OF THUMB 2

Deletions Based on Missing Data

- Variables with as little as 15 percent missing data are candidates for deletion [15], but higher levels of missing data (20% to 30%) can often be remedied
- Be sure the overall decrease in missing data is large enough to justify deleting an individual variable or case
- Cases with missing data for dependent variable(s) typically are deleted to avoid any artificial increase in relationships with independent variables
- When deleting a variable, ensure that alternative variables, hopefully highly correlated, are available to represent the intent of the original variable
- Always consider performing the analysis both with and without the deleted cases or variables to identify any marked differences

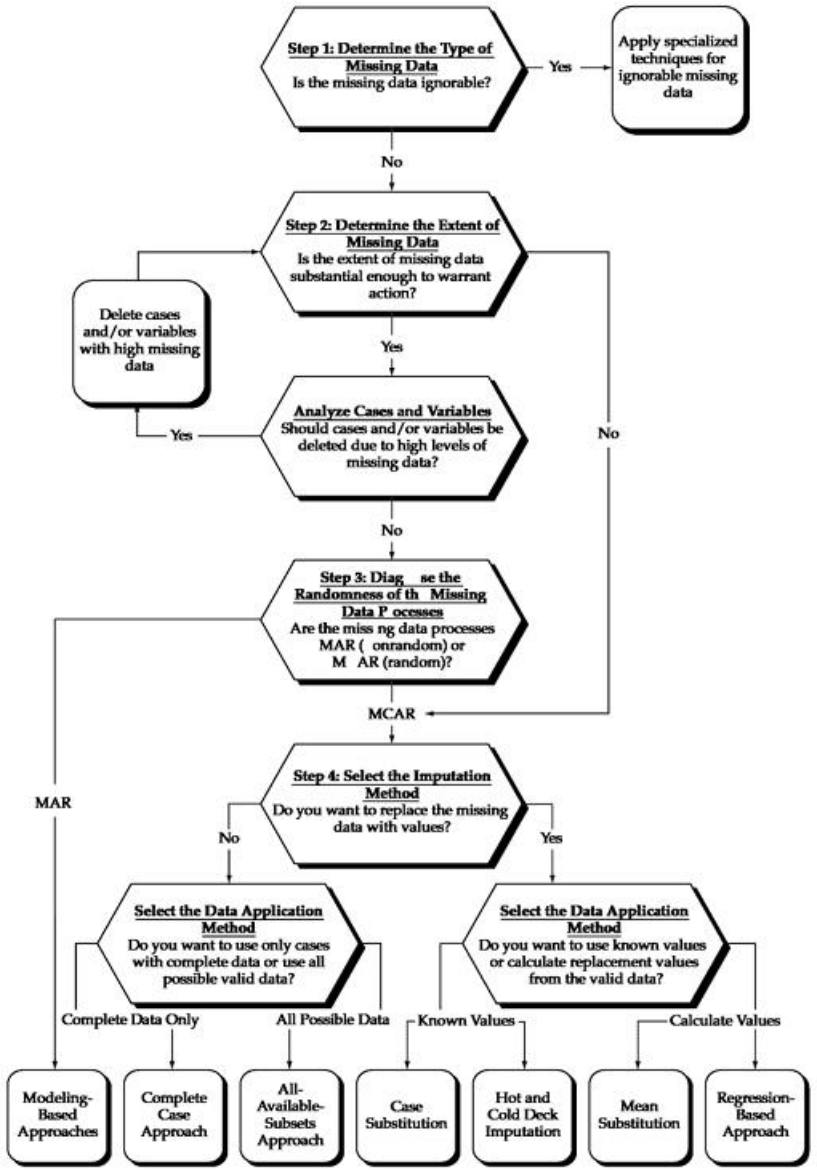


FIGURE 4 A Four-Step Process for Identifying Missing Data and Applying Remedies

3. Limpieza de Datos

Algunas “reglas” para la imputación de Missing

TABLE 2 Comparison of Imputation Techniques for Missing Data

Imputation Method	*Advantages*	*Disadvantages*	*Best Used When:*
		Imputation Using Only Valid Data	
Complete Data	• Simplest to implement • Default for many statistical programs	• Most affected by nonrandom processes • Greatest reduction in sample size • Lowers statistical power	• Large sample size • Strong relationships among variables • Low levels of missing data
All Available Data	• Maximizes use of valid data • Results in largest sample size possible without replacing values	• Varying sample sizes for every imputation • Can generate "out of range" values for correlations and eigenvalues	• Relatively low levels of miss ng data • Mode e relationships among variables
		Imputation Using Known Replacement Values	
Case Substitution	• Provides realistic replacement values (i.e., another actual observation) rather than calculated values	• Must have additional cases not in the original sample • Must define similari y measure to iden ify replacement c se	• Additional cases are available • Able to identify appropriate replacement cases
Hot and Cold Deck Imputation	• Replaces missing data with actual values from the most similar case or best known value	• Must de ine suitably simila ases or appropriate external valu s	• Established replacement values are known, or • Missing data process indicates variables upon which to base similarity
		Imputation by Calculating Replacement Values	
Mean Substitution	• Easily implemented • Provides all cases with complete information	• Reduces variance of the distribution • Distorts distribution of the data • Depresses observed correlations	• Relatively low levels of missing data • Relatively strong relationships among variables
Regression Imputation	• Em l ys actual ela onships among h variables • Replacement values calculated based on an observation's own values on other variables • Unique set of predictors can be used for each variable with missing data	• Reinforces existing relationships and reduces generalizability • Must have sufficient relationships among variables to generate valid predicted values • Understates variance unless error term added to replacement value • Replacement values may be "out of range"	• Moderate to high levels of missing data • Relationships sufficiently established so as to not impact generalizability • Software availability
		Model-Based Methods for MAR Missing Data Processes	
Model-Based Methods	• Accommodates both nonrandom and random missing data processes • Best representation of original distribution of values with least bias	• Complex model specification by researcher • Requires specialized software • Typically not available directly in software programs (except EM method in SPSS)	• Only method that can accommodate nonrandom missing data processes • High levels of missing data require least biased method to ensure generalizability

RULES OF THUMB 3

Imputation of Missing Data

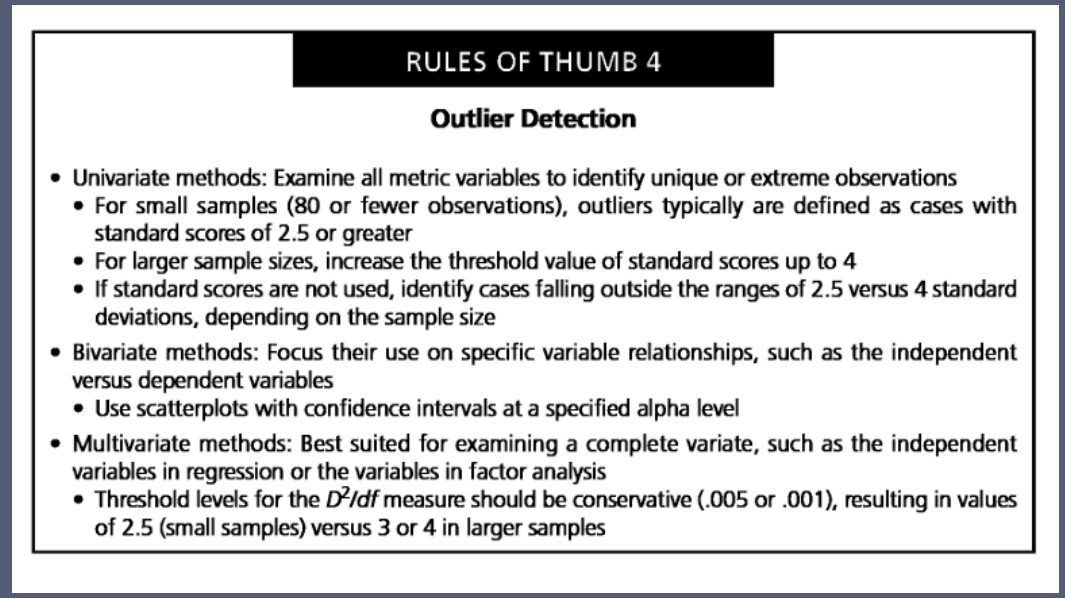
- **Under 10%** — Any of the imputation methods can be applied when missing data are this low, although the complete case method has been shown to be the least preferred
- **10% to 20%** — The increased presence of missing data makes the all-available, hot deck case substitution, and regression methods most preferred for MCAR data, whereas model-based methods are necessary with MAR missing data processes
- **Over 20%** — If it is deemed necessary to impute missing data when the level is over 20 percent, the preferred methods are:
  - The regression method for MCAR situations
  - Model-based methods when MAR missing data occur

3. Limpieza de Datos

El tratamiento de outliers es complejo.

Podemos definir un umbral para los valores atípicos. Esto requiere primero estandarizar los datos (media 0 y desvío 1).

Usamos Métodos Univariados, Bivariados, y Multivariados



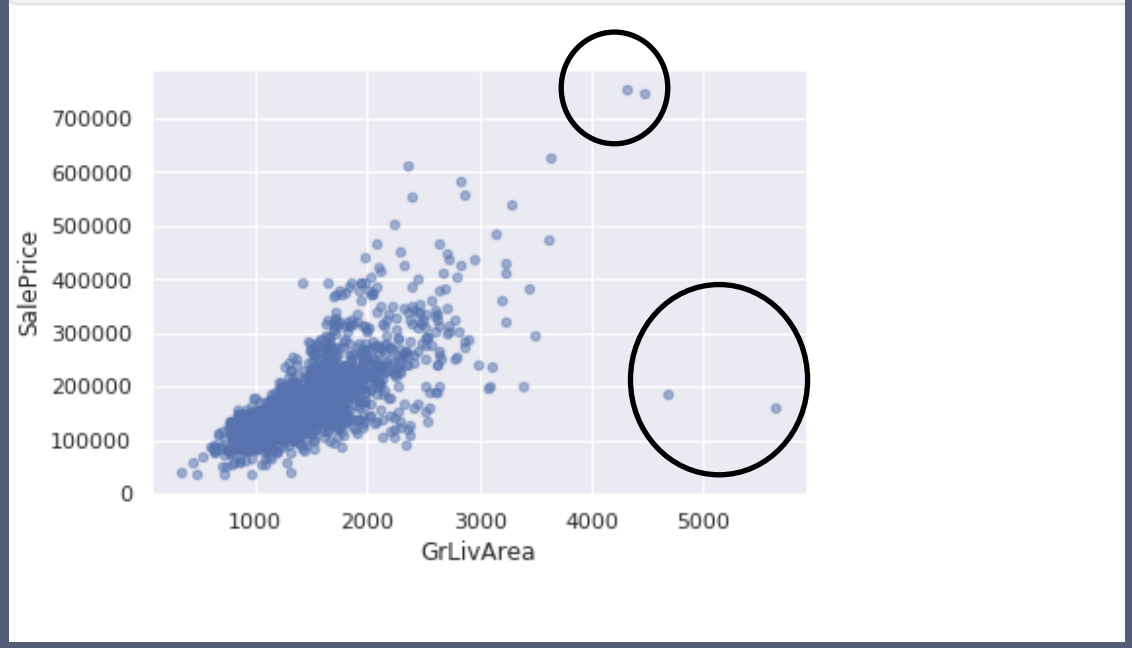
RULES OF THUMB 4

Outlier Detection

- Univariate methods: Examine all metric variables to identify unique or extreme observations
  - For small samples (80 or fewer observations), outliers typically are defined as cases with standard scores of 2.5 or greater
  - For larger sample sizes, increase the threshold value of standard scores up to 4
  - If standard scores are not used, identify cases falling outside the ranges of 2.5 versus 4 standard deviations, depending on the sample size
- Bivariate methods: Focus their use on specific variable relationships, such as the independent versus dependent variables
  - Use scatterplots with confidence intervals at a specified alpha level
- Multivariate methods: Best suited for examining a complete variate, such as the independent variables in regression or the variables in factor analysis
  - Threshold levels for the D^2/df measure should be conservative (.005 or .001), resulting in values of 2.5 (small samples) versus 3 or 4 in larger samples

3. Limpieza de Datos

```
# Análisis bivariable SalePrice/GrLivArea:

var = 'GrLivArea'
data = pd.concat([df_train['SalePrice'], df_train[var]], axis = 1)
data.plot.scatter(x = var, y = 'SalePrice', alpha = 0.5);
```

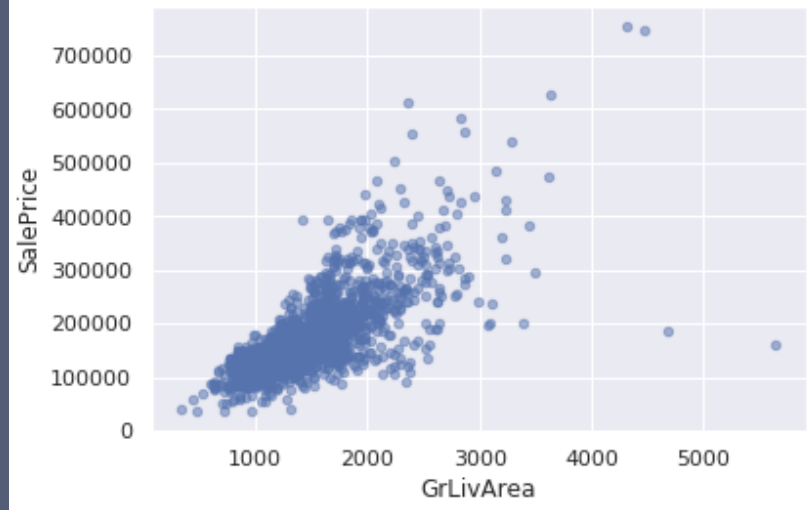


- ¿qué nos dicen estos posibles outliers?

- Debemos seguir analizando el resto de las variables más críticas (correlacionadas con la Y)

3. Limpieza de Datos

```
# Análisis bivariable SalePrice/GrLivArea:

var = 'GrLivArea'
data = pd.concat([df_train['SalePrice'], df_train[var]], axis = 1)
data.plot.scatter(x = var, y = 'SalePrice', alpha = 0.5);
```



- ¿qué nos dicen estos posibles outliers?

4. Normalización

'SalePrice' ¿cumple las asunciones estadísticas que nos permiten aplicar la regresión?
El resto de las variables, ¿presentan valores lineales?

Condiciones estadísticas sobre la Y (Variable a Predecir)

1)Normalidad: Los datos deben parecerse a una distribución normal. Si resolvemos la normalidad evitamos otros problemas, como la homocedasticidad.

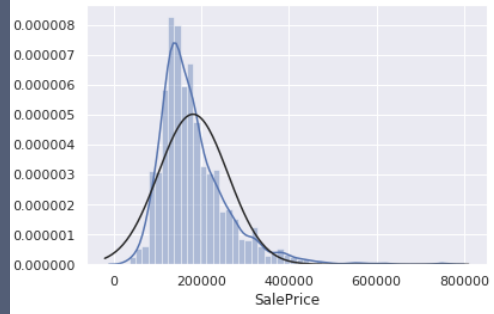
2) Homocedasticidad: La homocedasticidad se refiere a la suposición de que las variables dependientes tienen el mismo nivel de varianza en todo el rango de las variables predictoras. Es deseable porque queremos que el término de error sea el mismo en todos los valores de las variables independientes.

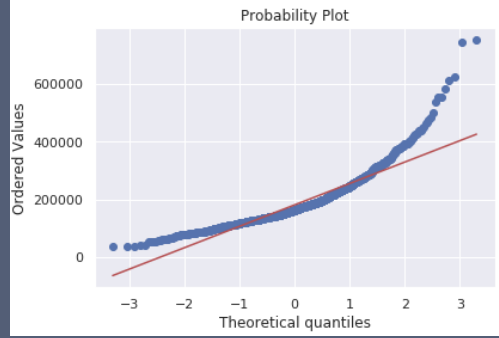
3) Linealidad: Miramos los diagramas de dispersión y buscar patrones lineales. Si los patrones no son lineales, conviene realizar las transformaciones a los datos. transformaciones de datos.

4) Ausencia de errores correlacionados - Ocurre en series temporales. Valores correlacionados con el tiempo.

4. Normalización

```
# Histograma y gráfico de probabilidad normal:

sns.distplot(df_train['SalePrice'], fit = norm);
fig = plt.figure()
res = stats.probplot(df_train['SalePrice'], plot = plt)
```



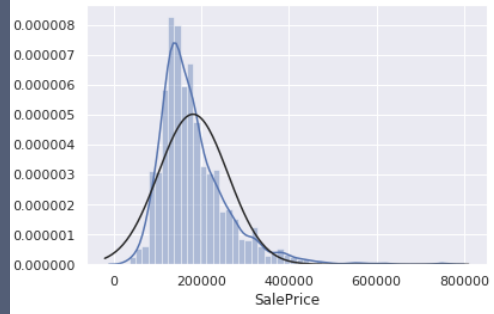


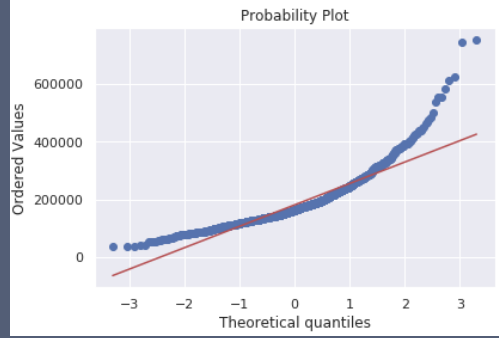
'SalePrice' vimos que no tenia distribución normal.

Una transformación, puede ayudarnos.

4. Normalización

```
# Histograma y gráfico de probabilidad normal:

sns.distplot(df_train['SalePrice'], fit = norm);
fig = plt.figure()
res = stats.probplot(df_train['SalePrice'], plot = plt)
```

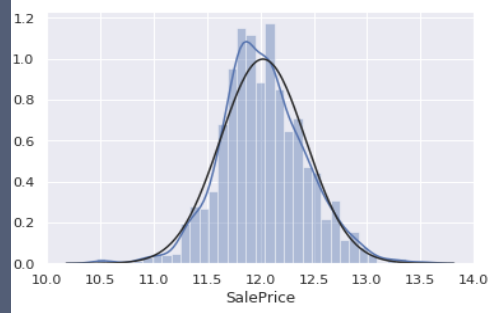


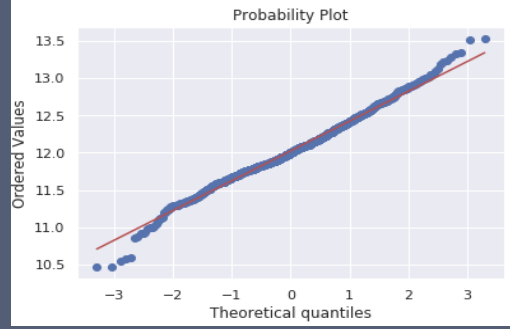


4. Normalización

```
df_train['SalePrice'] = np.log(df_train['SalePrice'])
```

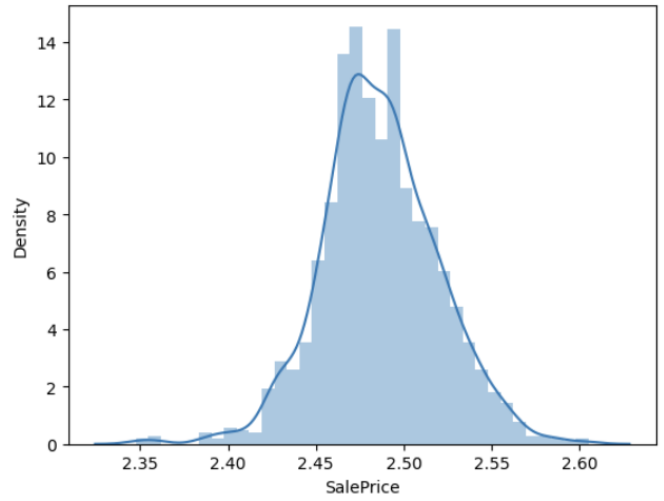
```
sns.distplot(df_train['SalePrice'], fit = norm);
fig = plt.figure()
res = stats.probplot(df_train['SalePrice'], plot = plt)
```






```
sns.distplot(np.log(X['SalePrice'])) #simetrica, cierta normalidad! Mucho mejor
X['SalePrice'] = np.log(X['SalePrice'])
```

```
<ipython-input-20-180b4aa3262f>:7: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  sns.distplot(np.log(X['SalePrice'])) #simetrica, cierta normalidad! Mucho mejor
```



4. Normalización

```
# Estandarización de datos:

saleprice_scaled = StandardScaler().fit_transform(df_train['SalePrice'][:,np.newaxis]);
low_range = saleprice_scaled[saleprice_scaled[:,0].argsort()][:10]
high_range = saleprice_scaled[saleprice_scaled[:,0].argsort()][-10:]
print('Fuera de la distribución (por debajo):')
print(low_range)
print('\nFuera de la distribución (por arriba):')
print(high_range)
```

```
Fuera de la distribución (por debajo):
[[-1.83820775]
 [-1.83303414]
 [-1.80044422]
 [-1.78282123]
 [-1.77400974]
 [-1.62295562]
 [-1.6166617 ]
 [-1.58519209]
 [-1.58519209]
 [-1.57269236]]

Fuera de la distribución (por arriba):
[[3.82758058]
 [4.0395221 ]
 [4.49473628]
 [4.70872962]
 [4.728631  ]
 [5.06034585]
 [5.42191907]
 [5.58987866]
 [7.10041987]
 [7.22629831]]
```

Buscamos valores que escapan del rango:

Estandarizamos primero (media=0, desvío std=1)

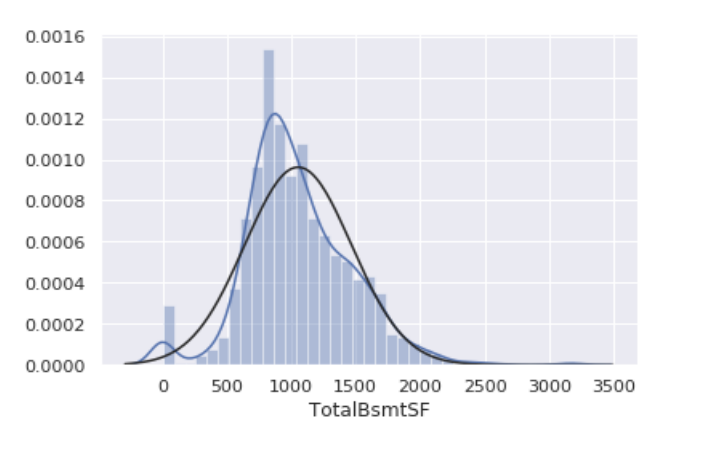
Hay valores que son bajos y similares y no muy alejados del 0.

Hay valores altos están muy alejados del 0.

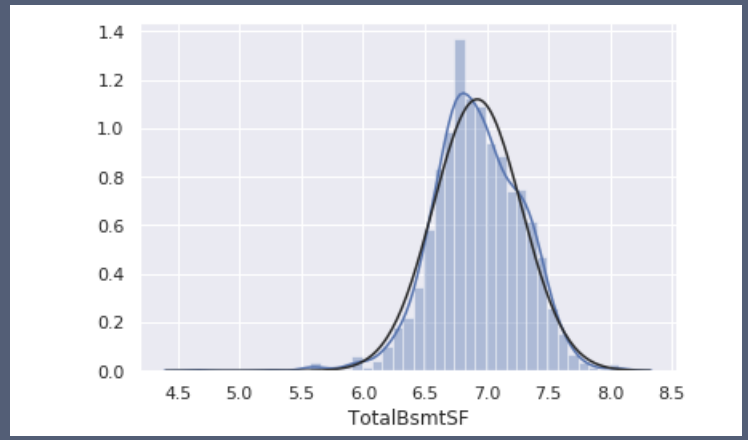
Los valores superiores a 7 están muy por fuera del rango.

4. Normalización

Antes de la transformación



Después de la transformación



Seguimos normalizando para el resto de las variables.

Antes busco:
Asimetrías, Sesgos,
Si hay muchos valores en cero (esto los tengo que evitar si transformo a logarítmica*).

*Se crea una dummy, binaria, y sobre los 1 se puede hacer el log.

5. Normalización

Buscamos homocedasticidad

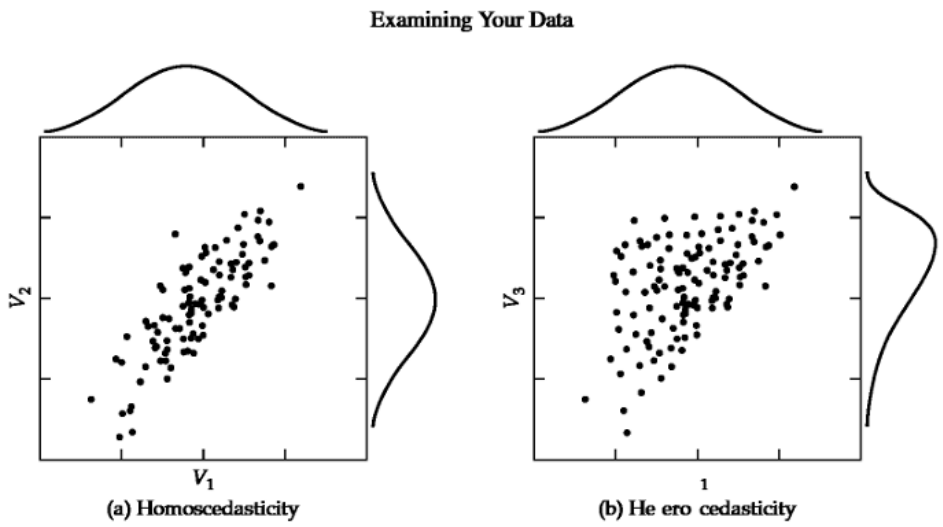


FIGURE 7 Scatterplots of Homoscedastic and Heterosceda tic Relationships

RULES OF THUMB 6

Transforming Data

- To judge the potential impact of a transformation, calculate the ratio of the variable's mean to its standard deviation:
  - Noticeable effects should occur when the ratio is less than 4
  - When the transformation can be performed on either of two variables, select the variable with the smallest ratio
- Transformations should be applied to the independent variables except in the case of heteroscedasticity
- Heteroscedasticity can be remedied only by the transformation of the dependent variable in a dependence relationship; if a heteroscedastic relationship is also nonlinear, the dependent variable, and perhaps the independent variables, must be transformed
- Transformations may change the interpretation of the variables; for example, transforming variables by taking their logarithm translates the relationship into a measure of proportional change (elasticity); always be sure to explore thoroughly the possible interpretations of the transformed variables
- Use variables in their original (untransformed) format when profiling or interpreting results

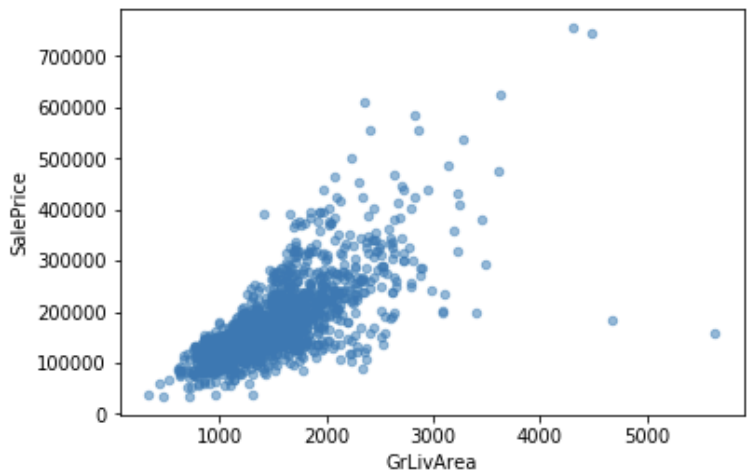
RULES OF THUMB 5

Testing Statistical Assumptions

- Normality can have serious effects in small samples (fewer than 50 cases), but the impact effectively diminishes when sample sizes reach 200 cases or more
- Most cases of heteroscedasticity are a result of nonnormality in one or more variables; thus, remedying normality may not be needed due to sample size, but may be needed to equalize the variance
- Nonlinear relationships can be well defined, but seriously understated unless the data are transformed to a linear pattern or explicit model components are used to represent the nonlinear portion of the relationship
- Correlated errors arise from a process that must be treated much like missing data; that is, the researcher must first define the causes among variables either internal or external to the dataset; if they are not found and remedied, serious biases can occur in the results, many times unknown to the researcher

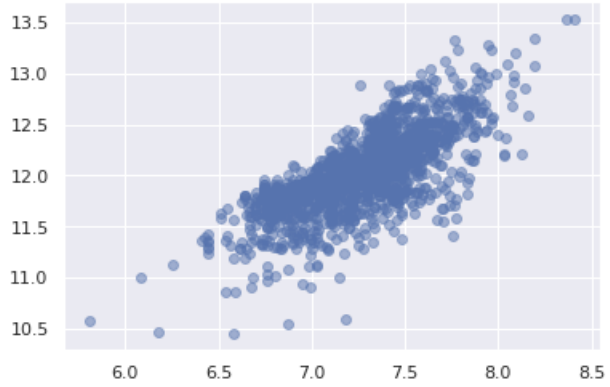
4. Normalización

Antes de la transformación



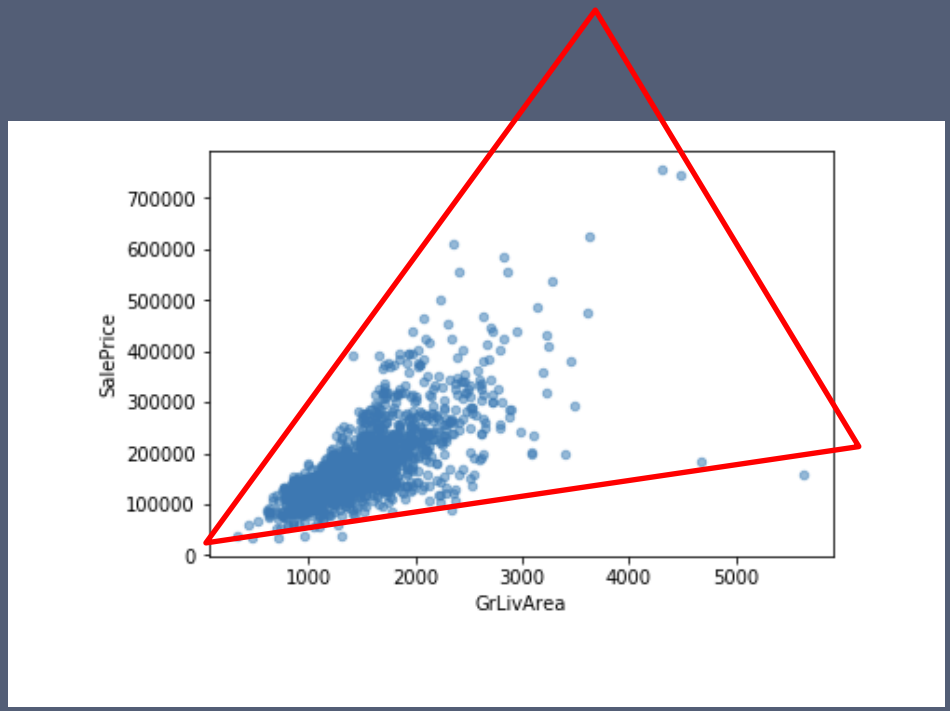
Después de la transformación

```
plt.scatter(df_train['GrLivArea'], df_train['SalePrice'], alpha = 0.5);
```



4. Normalización

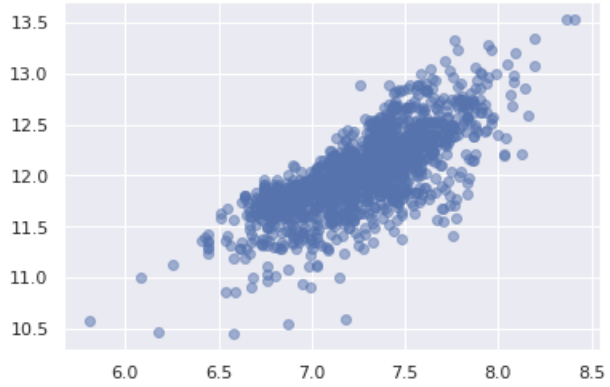
Antes de la transformación



Presencia de heterocedasticidad

Después de la transformación

```
plt.scatter(df_train['GrLivArea'], df_train['SalePrice'], alpha = 0.5);
```



Anexos

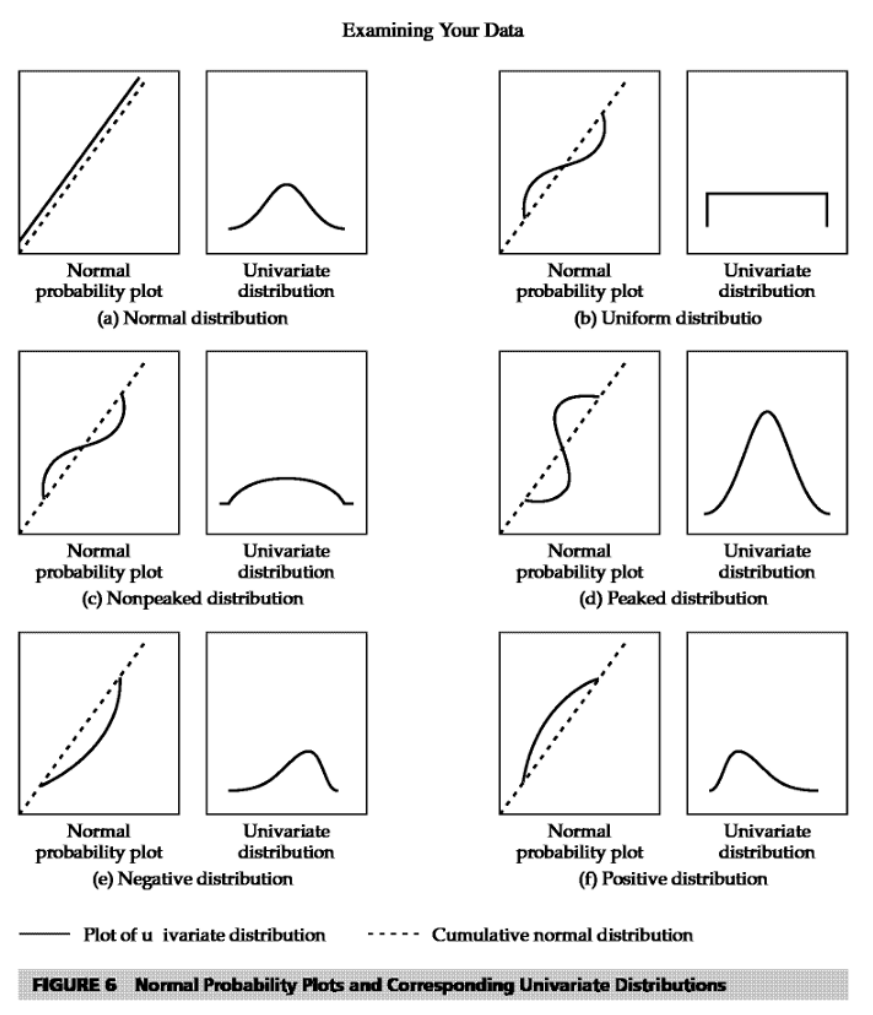


FIGURE 6 Normal Probability Plots and Corresponding Univariate Distributions

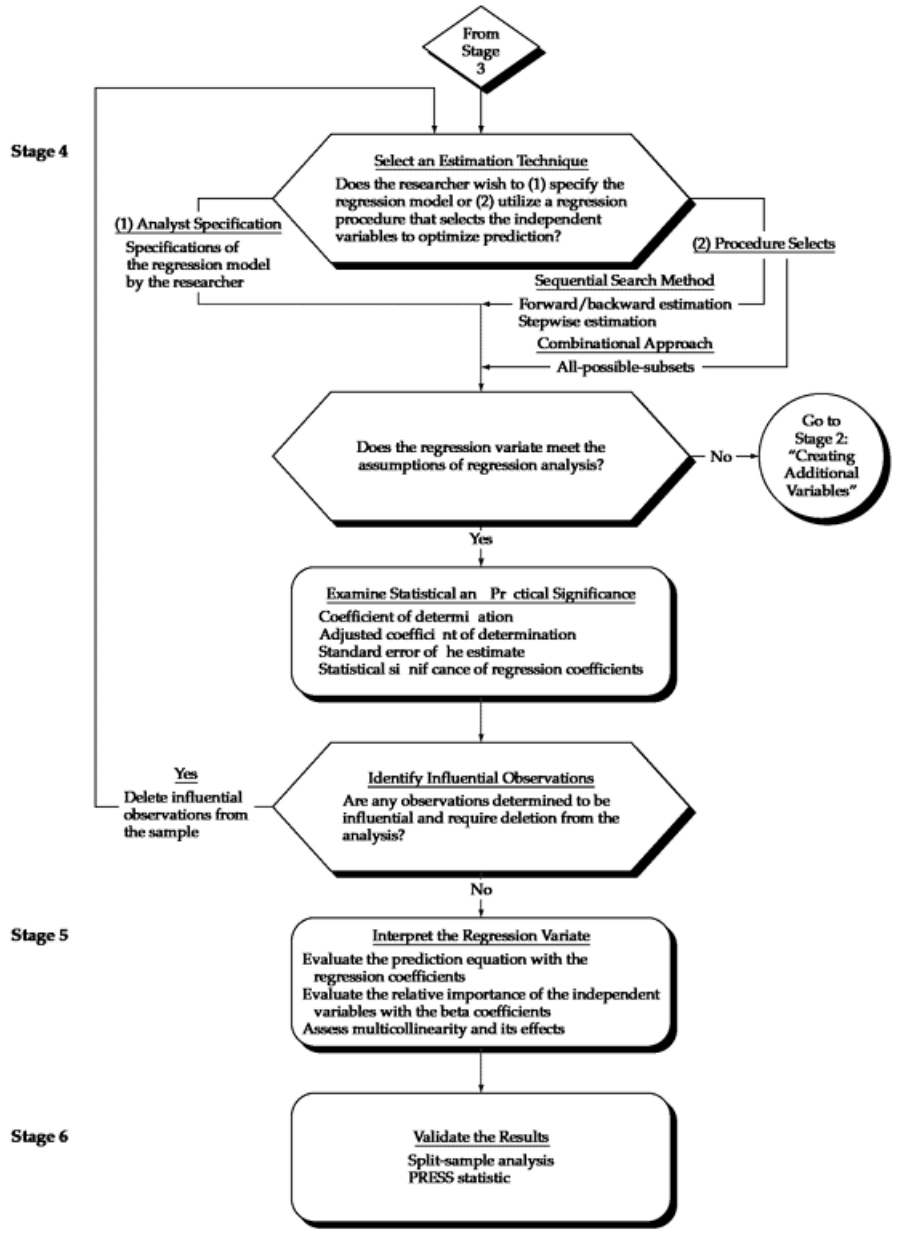


FIGURE 6 Stages 4–6 in the Multiple Regression Decision Diagram