DSCI 554 LECTURE 9

STATISTICS REVIEW, STATISTICAL GRAPHICS

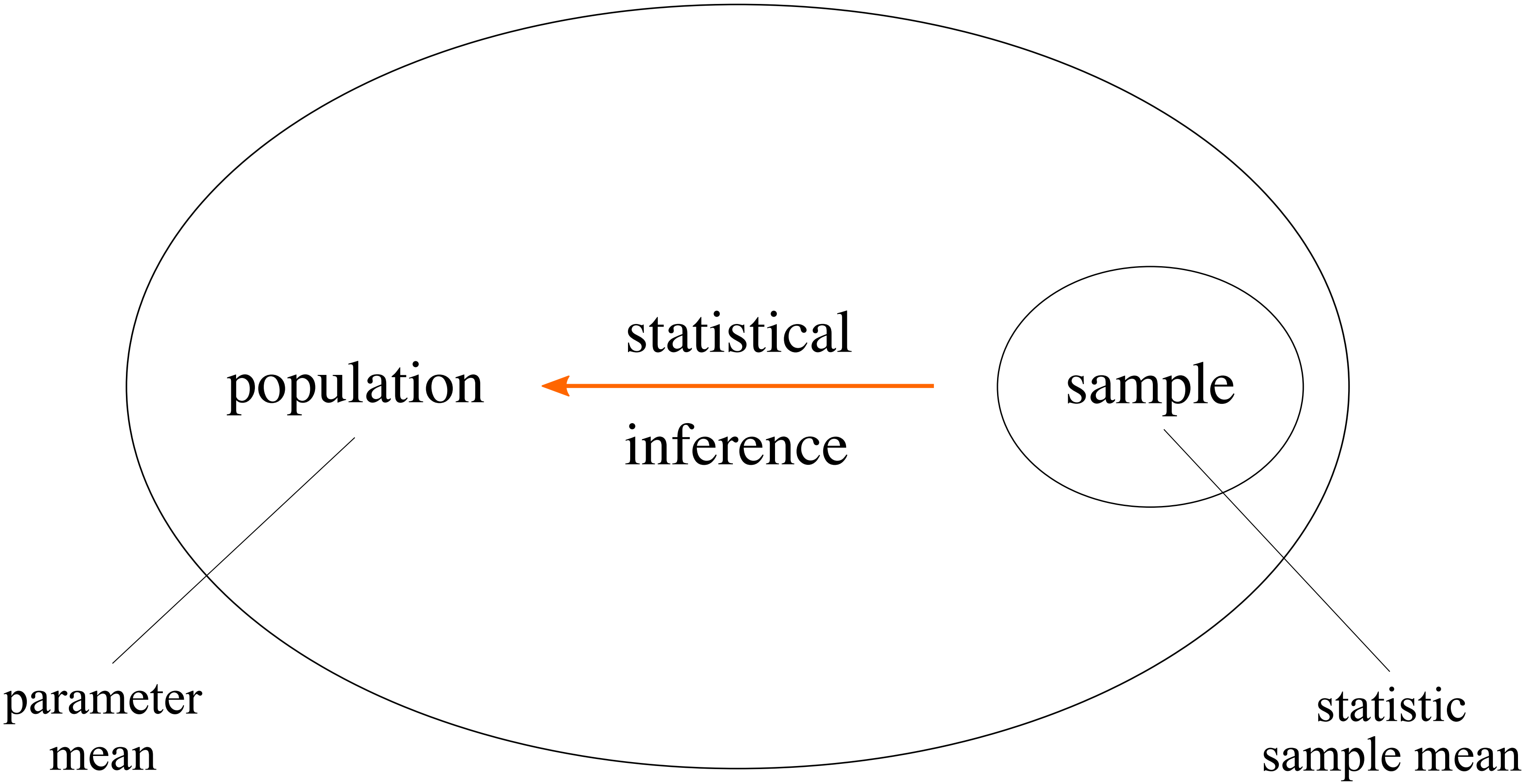
Dr. Luciano Nocera

USC Viterbi
School of Engineering
Integrated Media Systems Center

OUTLINE

- Basics of statistics and modeling
- Statistical graphics
- Tools

STATISTICS


statistical
population
inference
sample
parameter
mean
statistic
sample mean

TYPES OF STATISTICS

- **Descriptive statistics:** summarize the data, i.e. one number stands for a group of numbers

  Examples: mean, median, SD

- **Inferential statistics:** infer (model) population data from sample data

  Examples: hypothesis testing, regression analysis

DATA NOMENCLATURE

	ML	**Stats**
Observations	Samples	Cases
Attribute	Feature	Independent variable
Class	Label	Dependent variable

INDEPENDENT VS. DEPENDENT VARIABLES

Statistics

$$\text{dependent variable} = f(\text{independent variables})$$

Machine learning

$$\text{label} = f(\text{features})$$

INDEPENDENT AND DEPENDENT VARIABLES EXAMPLES

Height depends on age

Time spent studying affects test score

Medication in persons with Parkinson's Disease affects the SD of the step length

MEASURES OF ORDER

K^th order statistic: value at position k in ordered data

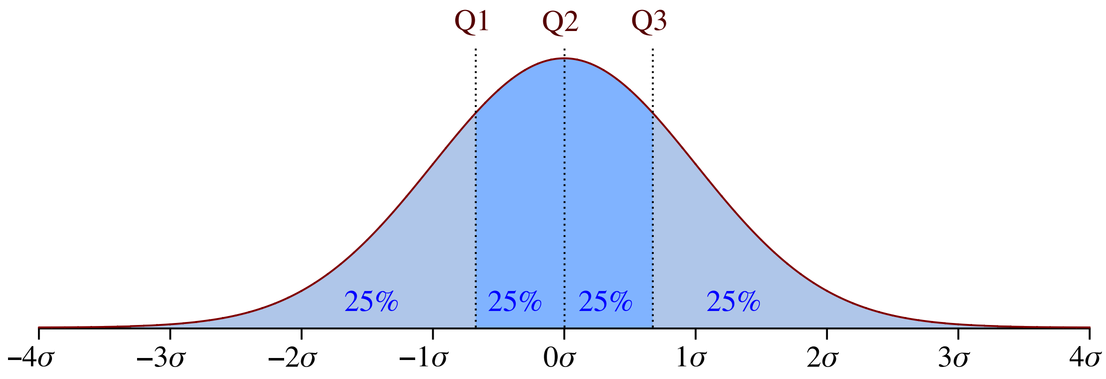
Range: range of values

Modes/peaks: most frequent values

$$\text{data} = [X_1, \ldots, X_N] = [0, 1, 1, 2, 2, 3, 4, 15]$$
$$1^{st}\text{order: } X_1 = \min(X_1, \ldots, X_N) = 0$$
$$N^{st}\text{order: } X_N = \max(X_1, \ldots, X_N) = 15$$
$$\text{range: } X_N - X_1 = 15$$
$$\text{modes: } \{1, 2\}$$

QUANTILES

- **q-quantiles** divide the observations in q groups using $q - 1$ values
- **Quartiles** divide the observations in 4 groups using 3 values:
  - Q_1: 25% at or below and 75% above
  - Q_2: 50% at or below and 50% above (median)
  - Q_3: 75% at or below and 25% above



Quartiles in a normal distribution [Ark0n derivative work: Gato ocioso]

$$\text{data} = [0, 1, 1, 2, 2, 3, 4, 15]$$
$$Q_1 = 1, Q_2 = 2, Q_3 = 3.25$$

Example based on SciPy formulation: $N = 8$ with N+1 parts. k-th q-quantile: $p = k/q, h = (N+1)p$, $x\lfloor h \rfloor + (h - \lfloor h \rfloor)(x\lfloor h \rfloor + 1 - x\lfloor h \rfloor)$

MEASURES OF CENTRAL TENDENCY

Median: value in the middle
Mean: sum divided by N

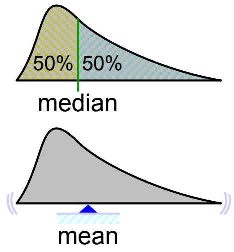
$$\mu = \bar{X} = \sum_{i=1}^{N} \frac{X_i}{N}$$

Standard deviation: dispersion

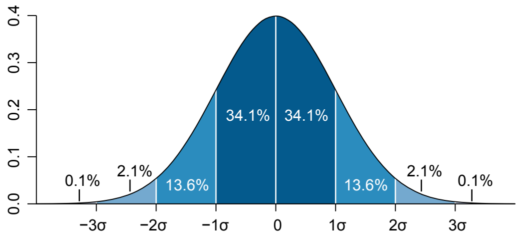
$$\sigma = \sqrt{\frac{1}{N-1} \sum_{i}^{N} (X_i - \bar{X})^2}$$

Variance: variation around the mean

$$\sigma^2$$



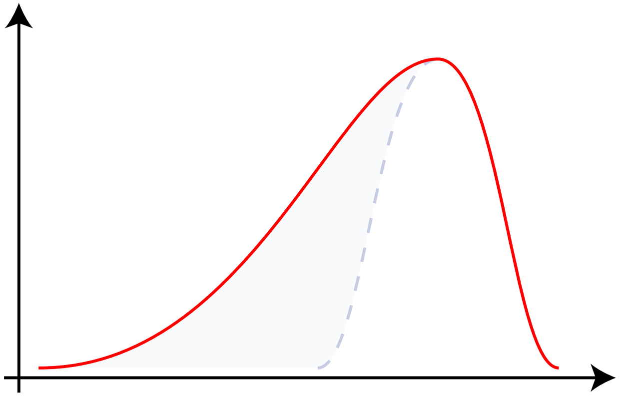
Median and mean (adapted from Cmglee - Own work)



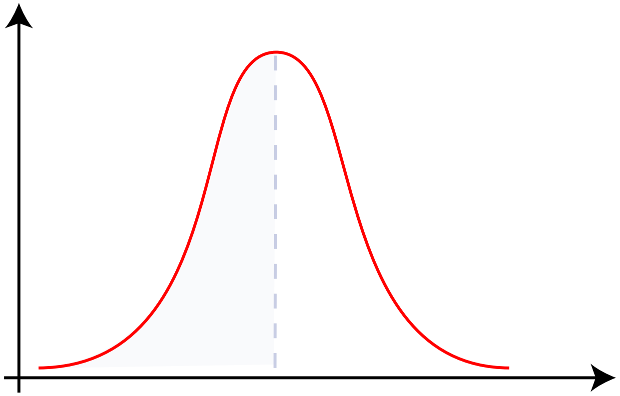
Normal distribution with bands of 1 σ (M. W. Toews - Own work)

$$\text{data} = [0, 1, 1, 2, 2, 3, 4, 15]$$
$$\text{median: } \tilde{X} = 2$$
$$\text{mean: } \bar{X} = 3.5$$
$$\text{standard deviation: } \sigma = 4.810702$$
$$\text{variance: } \sigma^2 = 23.142857$$

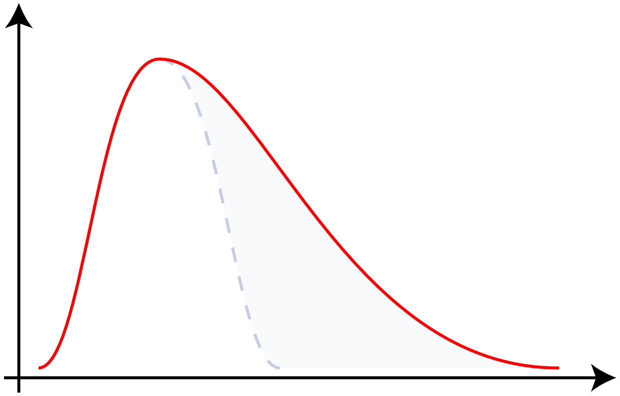
SKEWNESS



negative skew
left-skewed
left-tailed
skewed to the left



0 skewness
symmetric



positive skew
right-skewed
right-tailed
skewed to the right

FREQUENCY & RELATIVE FREQUENCY

Frequency: times event i occurs

$$n_i$$

Relative frequency: frequency normalized

$$f_i = \frac{n_i}{N}$$

$$\text{with } N = \sum_{k=1}^{K} n_k$$

$$\text{data} = [A, B, B, A, C, A, C, A]$$

$$n_A = 4, n_B = 2, n_c = 2$$

$$f_A = \frac{4}{8} = 0.5, f_B = \frac{2}{8} = 0.25, f_C = \frac{2}{8} = 0.25$$

$$N = n_A + n_B + n_c = 4 + 2 + 2 = 8$$

STATISTICS BY DATA TYPE

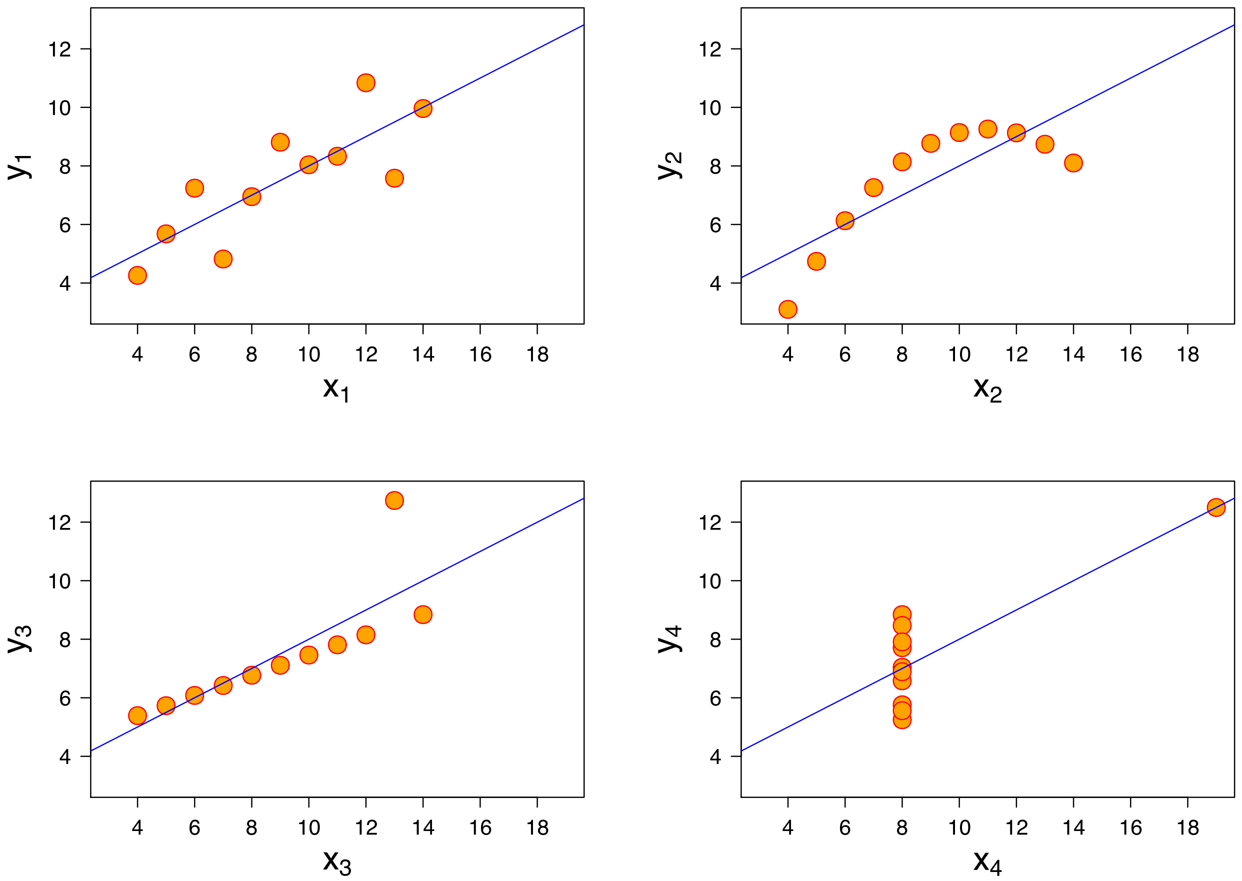
	Nominal	Ordinal	Interval	Ratio
Frequency	Yes	Yes	Yes	Yes
Median and percentile	No	Yes	Yes	Yes
Mean, SD, SEM*	No	No	Yes	Yes
Ratio, rate of variation	No	No	No	Yes

* standard error of the mean (SEM): $\sigma_{\bar{X}} = \frac{\sigma}{\sqrt{N}}$

OUTLINE

- Basics of statistics and modeling
- Statistical graphics
- Tools

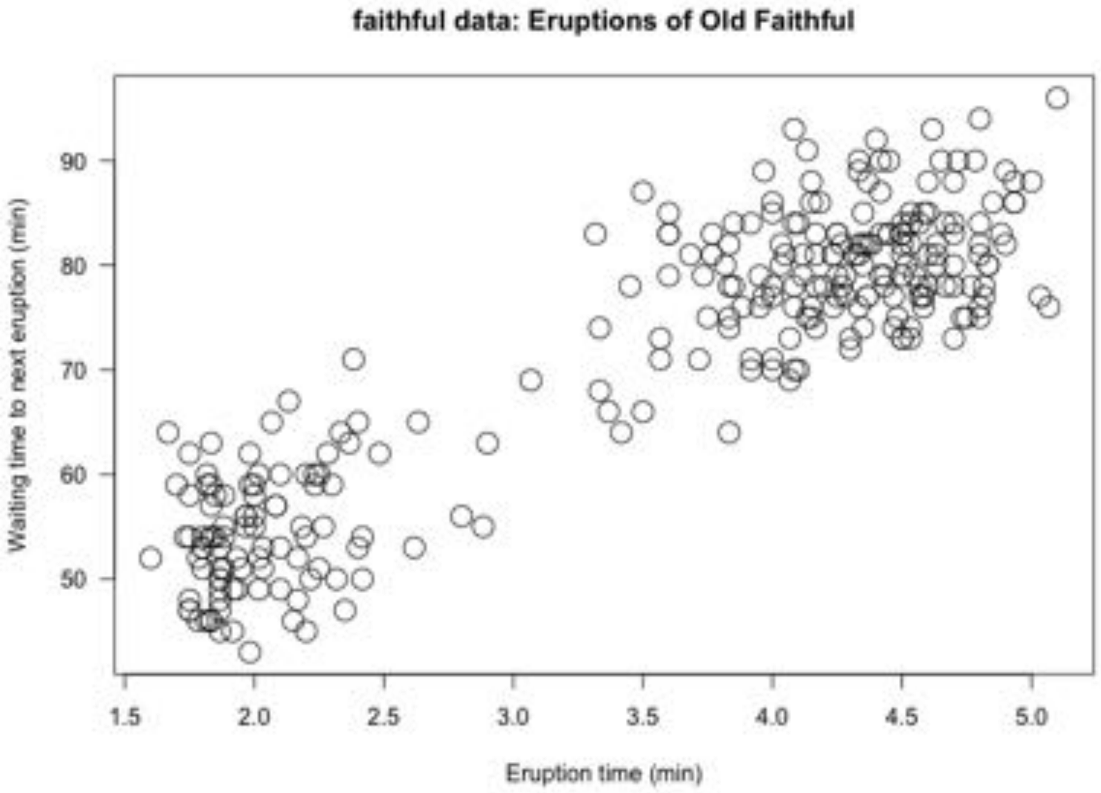
IMPORTANCE OF GRAPHING



Anscombe's quartet [Anscombe73] showing the importance of graphing before analysis

SCATTERPLOT

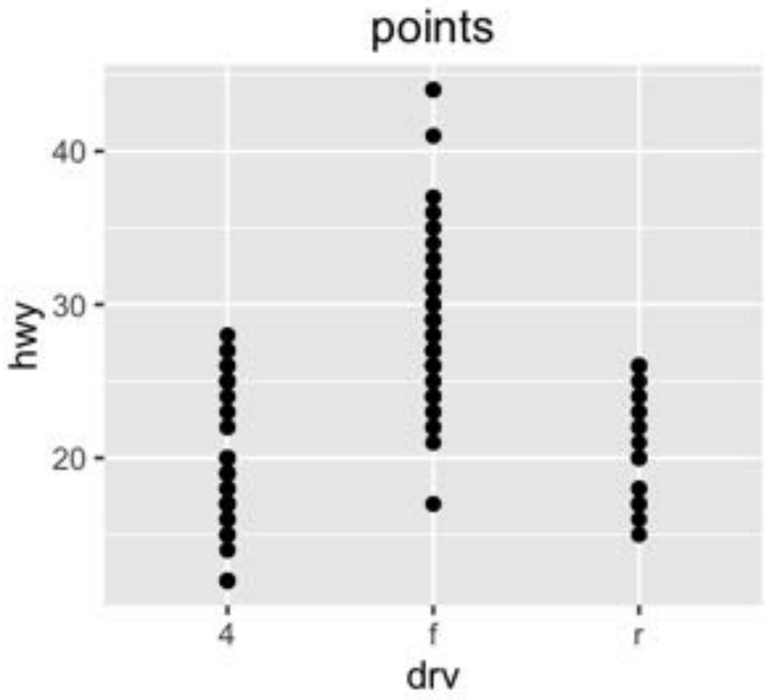
Shows distribution modes, skewness, outliers



Waiting time between eruptions and the duration of the eruption for the Old Faithful Geyser in Yellowstone National Park, Wyoming, USA.
The chart suggests there are two "types" of eruptions: short-wait-short-duration, and long-wait-long-duration.

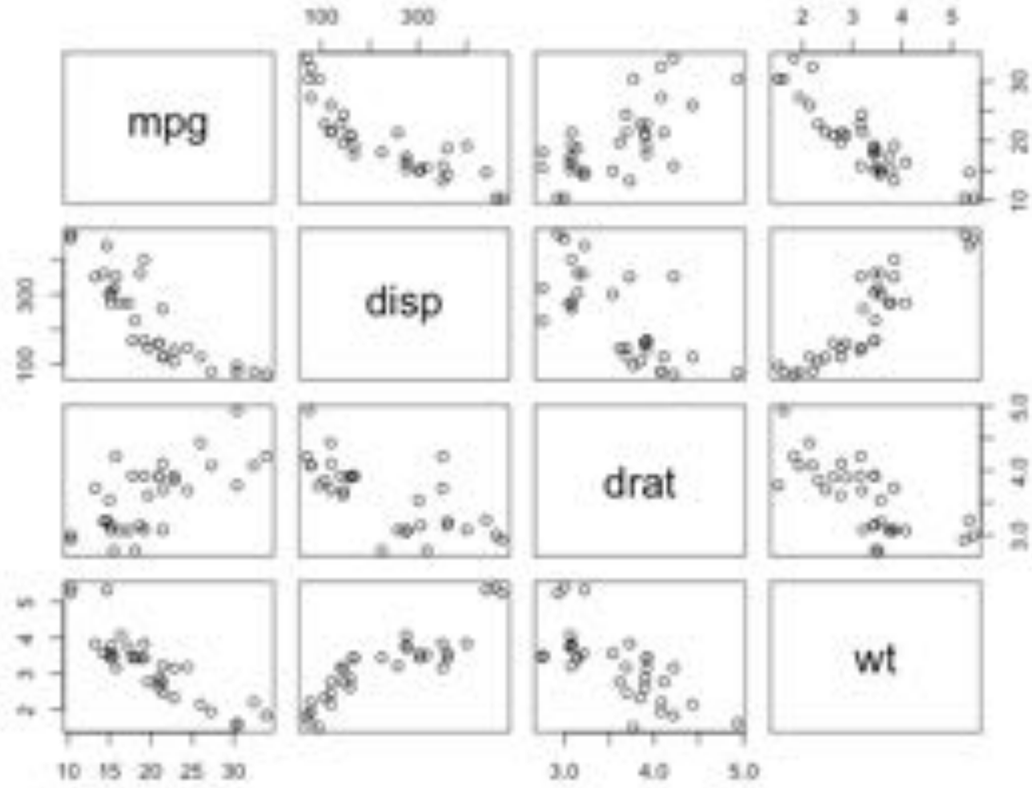
STRIPCHART (1D SCATTERPLOT)

Useful for comparing across categories



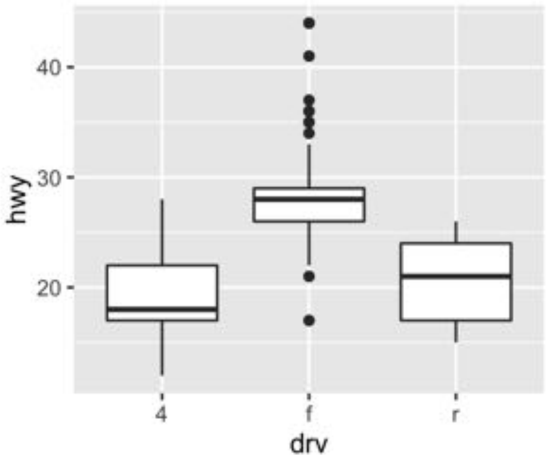
SCATTERPLOT MATRIX

Scatterplots of multivariate data

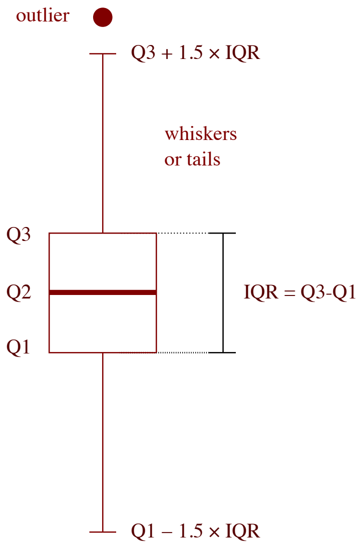


BOX-AND-WHISKER PLOT [TUCKEY 1969]

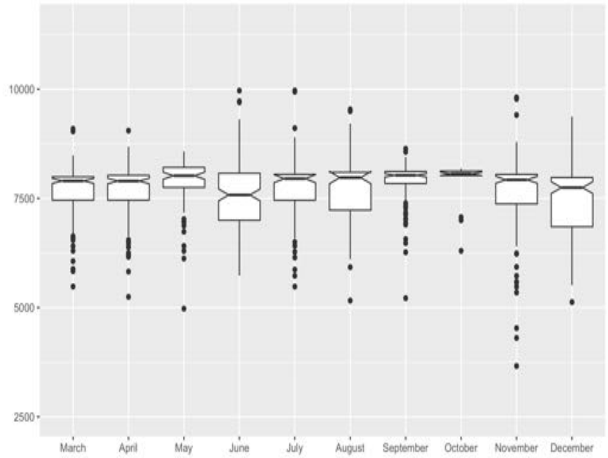
Boxplots visualize quartiles, distribution skewness, tails, outliers in unimodal distributions



Boxplots



Boxplot description



Boxplots with notches

FREQUENCY DISTRIBUTION TABLE

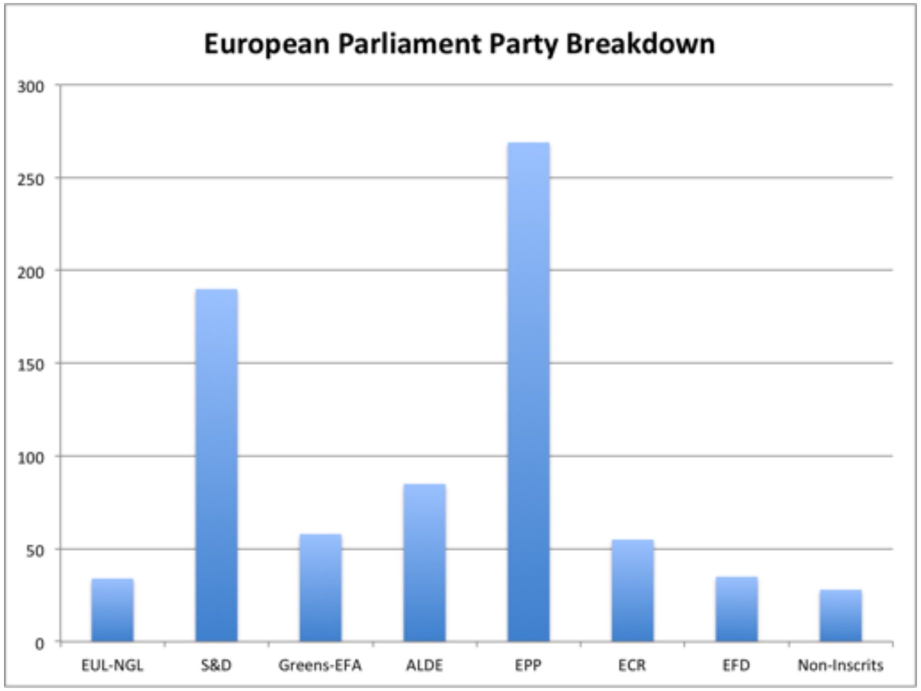
Often shown with ordered data, relative frequency and cumulative frequency

Chol. (mg/dl)	No.	Rel. Freq.	Cum. Freq.
80-119	13	1.2	1.2
120-159	150	14.1	15.3
160-199	442	41.4	56.7
200-239	299	28.0	84.7
240-279	115	10.8	95.5
280-319	34	3.2	98.7
320-359	9	0.8	99.5
360-399	5	0.5	100.0

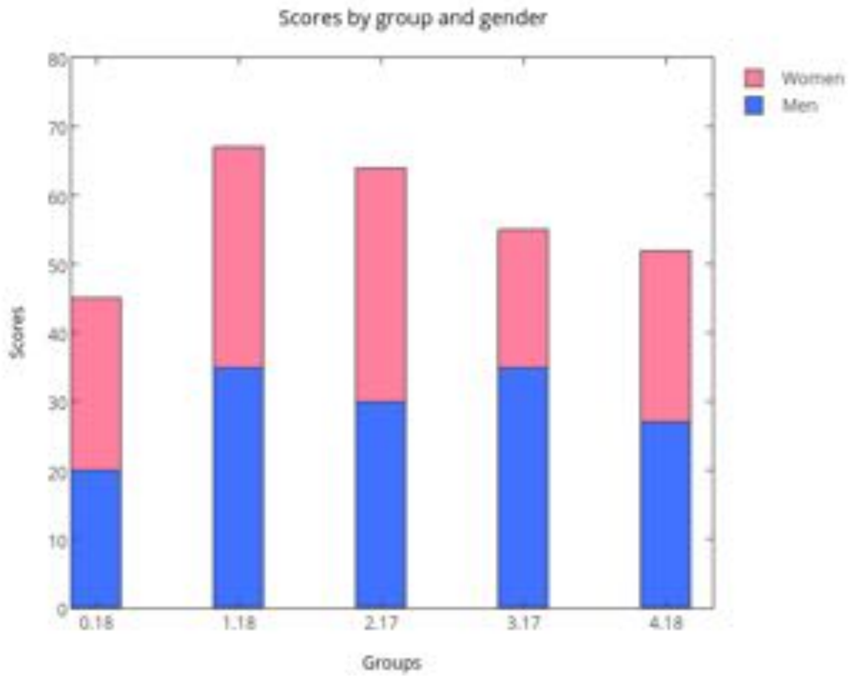
Frequencies of serum cholesterol levels for 1,067 US males, 25-34 years,1976-80

BAR CHARTS OF FREQUENCIES

Bars separation used to imply discontinuity



Bars for groups



Stacked bars for subgroups

STEM-AND-LEAF PLOT

Shows the data and the data distribution

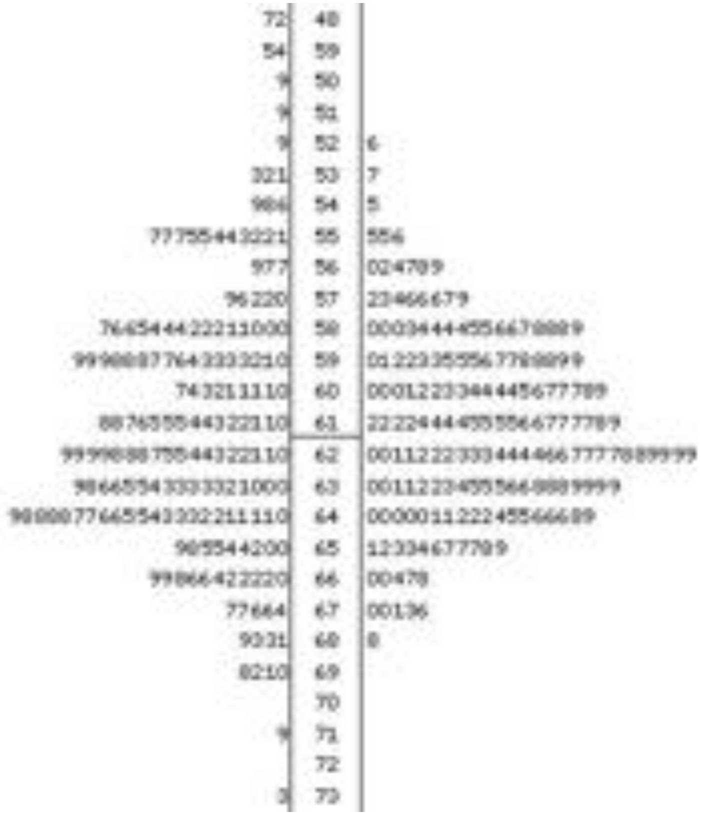


Figure 2. Distribution of cerebellar weights in the F2 intercross as illustrated by stem-and-leaf plots. The values on the left are the observed values, those on the right reflect correction by regression for brain weight. The mean for both distributions is marked by a horizontal line. Airey DC, Lu L, Williams RW Genetic control of the mouse cerebellum: identification of quantitative trait loci modulating size and architecture. J Neuroscience, 2001.

BUILDING A STEM-AND-LEAF PLOT

73, 42, 67, 78, 99, 84, 91, 82, 86, 122

BUILDING A STEM-AND-LEAF PLOT

73, 42, 67, 78, 99, 84, 91, 82, 86, 122

1. Order in ascending order

   42, 67, 73, 78, 82, 84, 86, 91, 99, 122

BUILDING A STEM-AND-LEAF PLOT

`73, 42, 67, 78, 99, 84, 91, 82, 86, 122`

1. Order in ascending order

   `42, 67, 73, 78, 82, 84, 86, 91, 99, 122`

2. Select stem and leaf

   `42, 67, 73, 78, 82, 84, 86, 91, 99, 122`

BUILDING A STEM-AND-LEAF PLOT

```
73, 42, 67, 78, 99, 84, 91, 82, 86, 122
```

1. Order in ascending order

    ```
    42, 67, 73, 78, 82, 84, 86, 91, 99, 122
    ```

2. Select stem and leaf

    ```
    42, 67, 73, 78, 82, 84, 86, 91, 99, 122
    ```

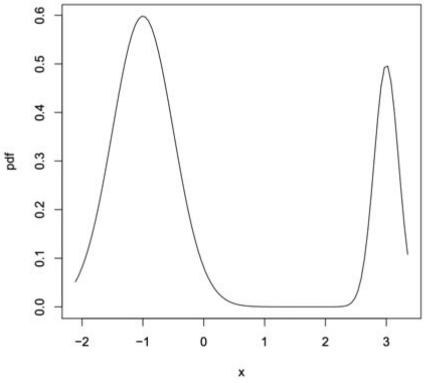
3. Plot

```
 4 | 2
 5 |
 6 | 7
 7 | 38
 8 | 246
 9 | 19
10 |
11 |
12 | 2
```

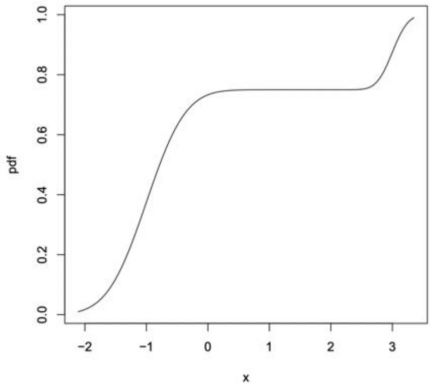
```
 4 | 2
 6 | 738
 8 | 24619
10 |
12 | 2
```

Half the size

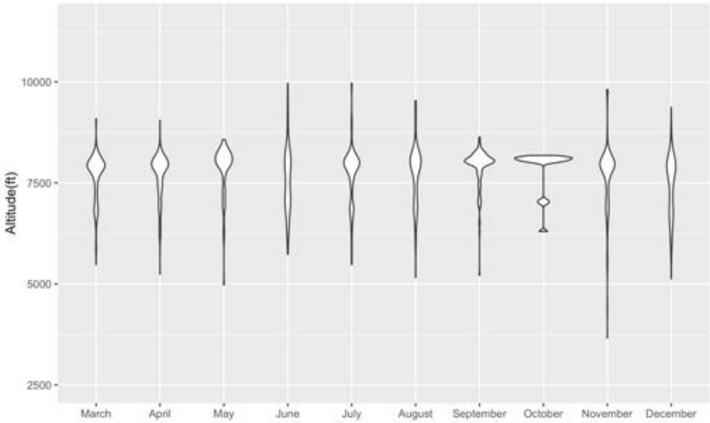
PDF & CDF PLOTS



Probability density plot



Cumulative density plot

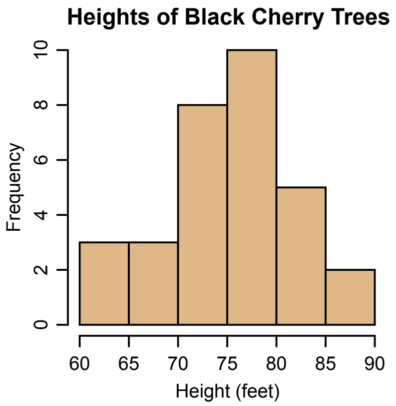


Violin plot: mirrored probability density plot

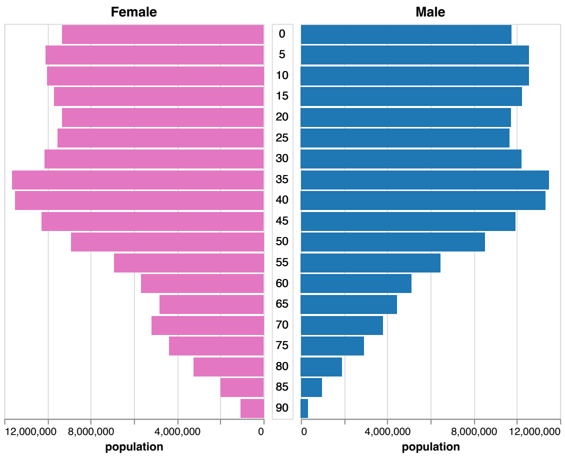
HISTOGRAM AND FREQUENCY POLYGON

Shows skewness, modes, tails, outliers

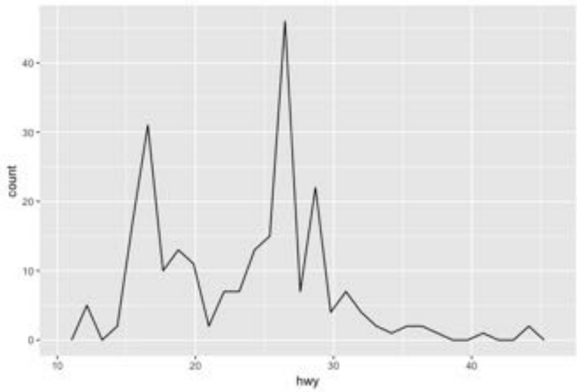
- Bar graph of frequencies for ordered, equal size bins
- Bars touch to imply continuity of bins
- Need to experiment with the bin size



Histogram [Pearson 1895] Black cherry tree histogram.svg from Wikimedia Commons



Population pyramid showing the distribution of age groups in a population. Stacked with shift at origin. Bars separation used to imply continuity.



Frequency polygon plot

BUILDING AN HISTOGRAM

```
73, 42, 67, 78, 99, 84, 91, 82, 86, 122
```

1. Order in ascending order

```
42, 67, 73, 78, 82, 84, 86, 91, 99, 122
```

2. Select bin size

```
range = max - min = 122 - 42 = 80
bin size 20
bin size 40
```

3. Create a **frequency table**

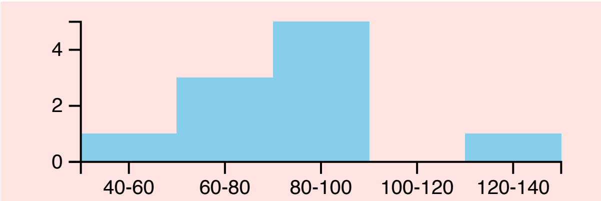
Interval	Frequency
40-60	1
60-80	3
80-100	5
100-120	0
120-140	1

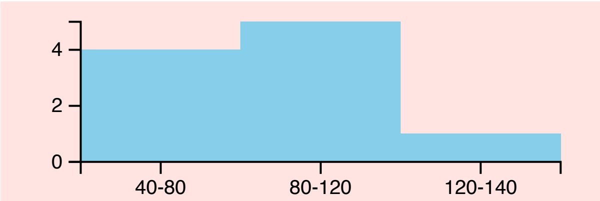
Bin size 20

Interval	Frequency
40-80	4
80-120	5
120-140	1

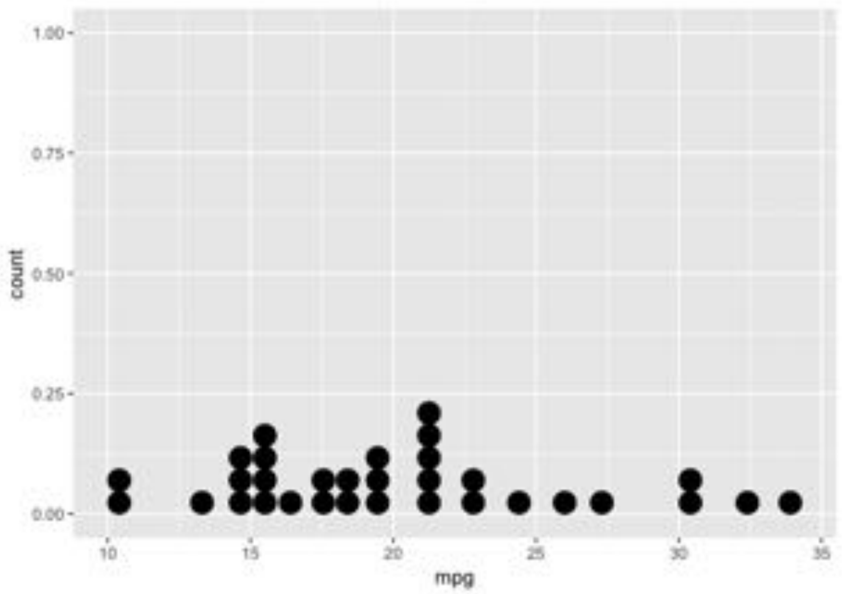
Bin size 40

4. Plot

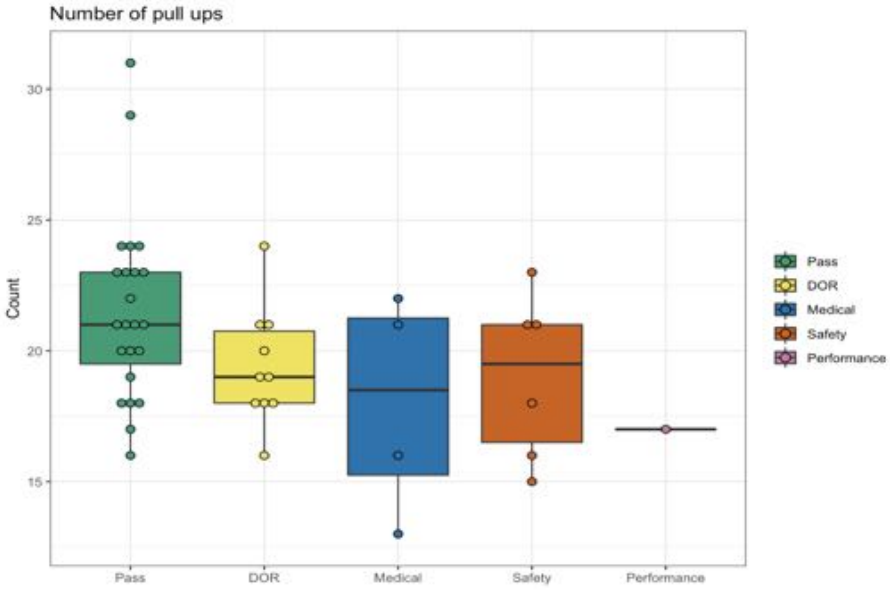




DOT PLOT



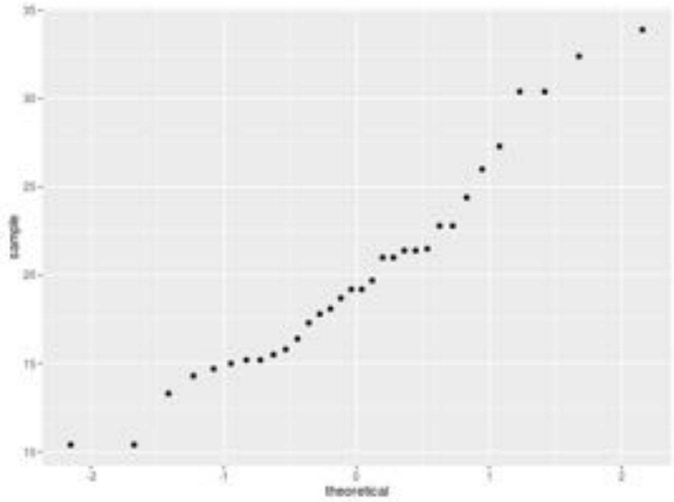
Dot plot histogram: y axis is the relative frequency, x axis is the dimension considered, each dot represents one observation, circle center is equal to the bin center, dot diameter is proportional (factor of 1 in the figure) to bin size.



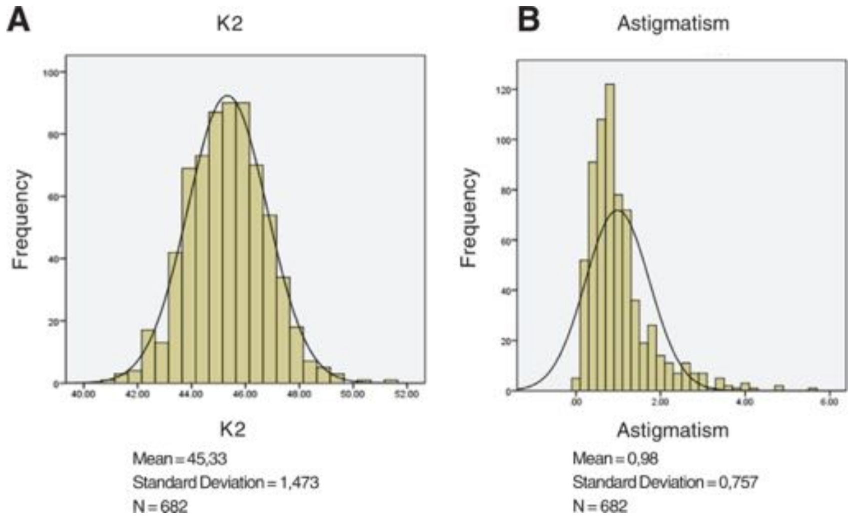
Boxplot with dotplot, each dot represents one observation

VISUALIZING NORMALITY

Q-Q plot and histograms



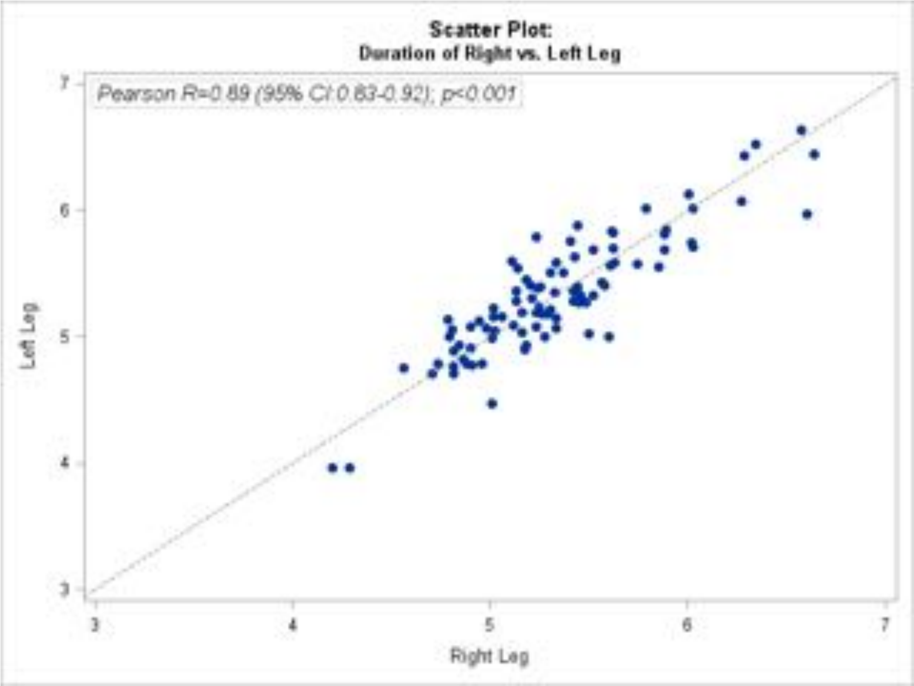
Q–Q (quantile-quantile) plot is a graphical method for comparing two probability distributions by plotting their quantiles against each other. Here we Assess normality by plotting against a normal distribution.



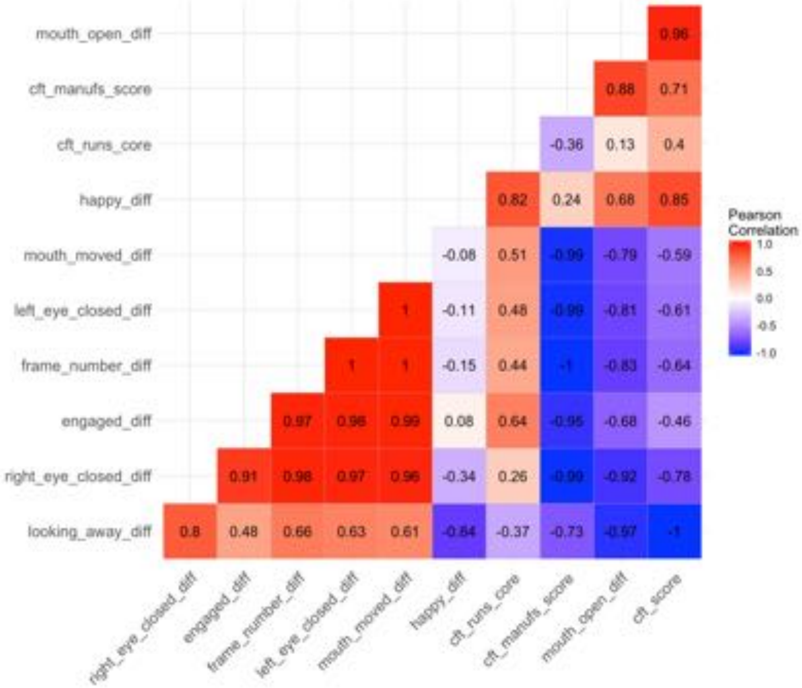
Histogram with superimposed line chart of normal distribution

VISUALIZING CORRELATIONS

Scatterplots and heatmaps



PCC* scatterplot and linear regression line.

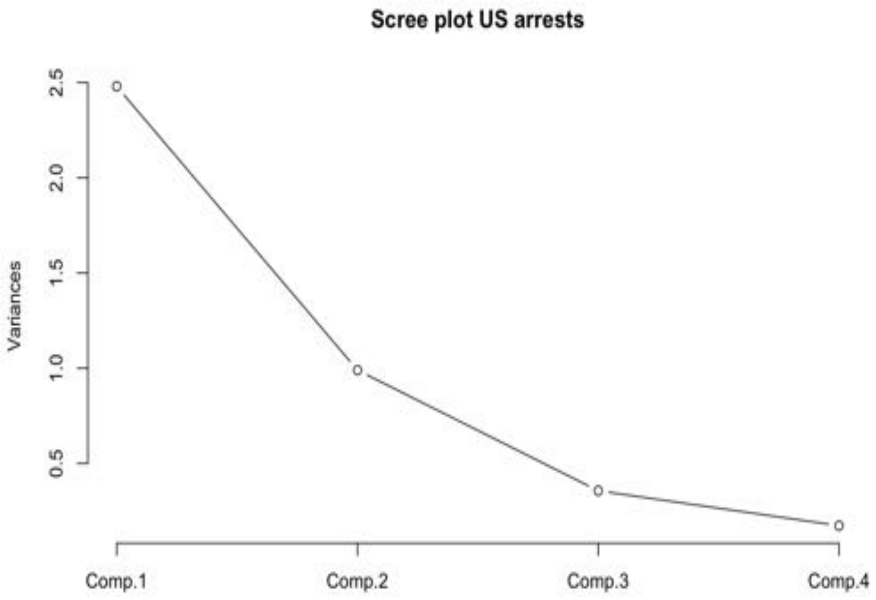


Heatmap of PCC* is a graphical tool to assess correlations in multivariate data. Note the diverging R-B color scale.

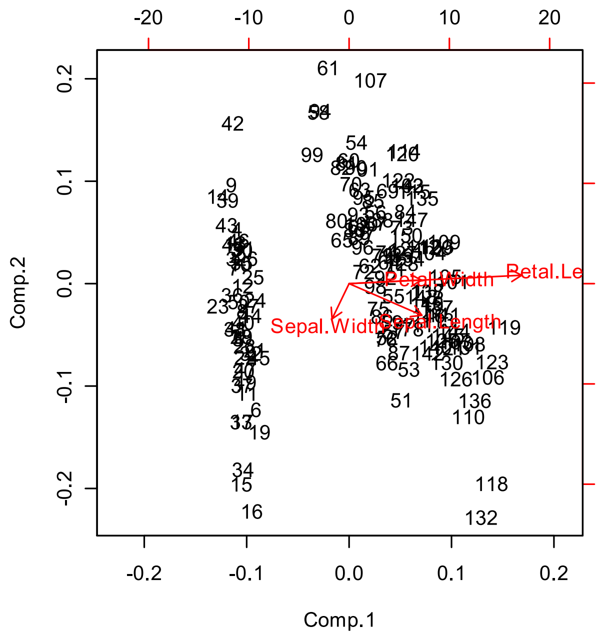
* Pearson's correlation coefficients (PCC) or Pearson's r, is a measure of linear correlation between two sets of data

VISUALIZING PCA RESULTS

Scree plot and Biplot



A scree plot shows how PCA* components explain data variability

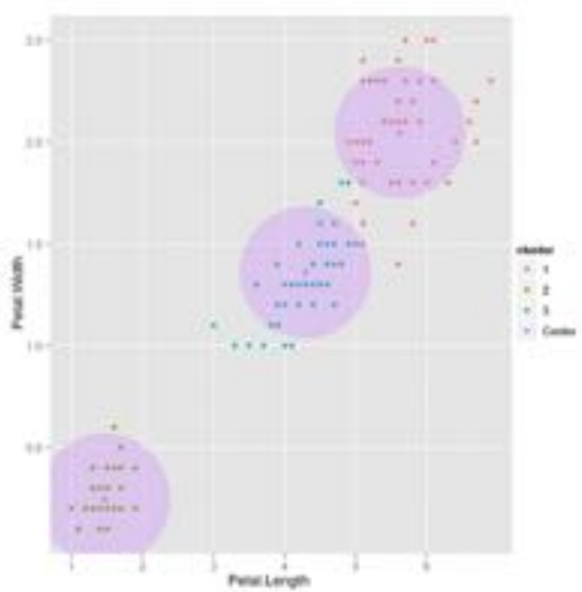


A Biplot [Gabriel 71] shows samples (points) and variables (vectors) with similar values plotted in the plane of PCA* components

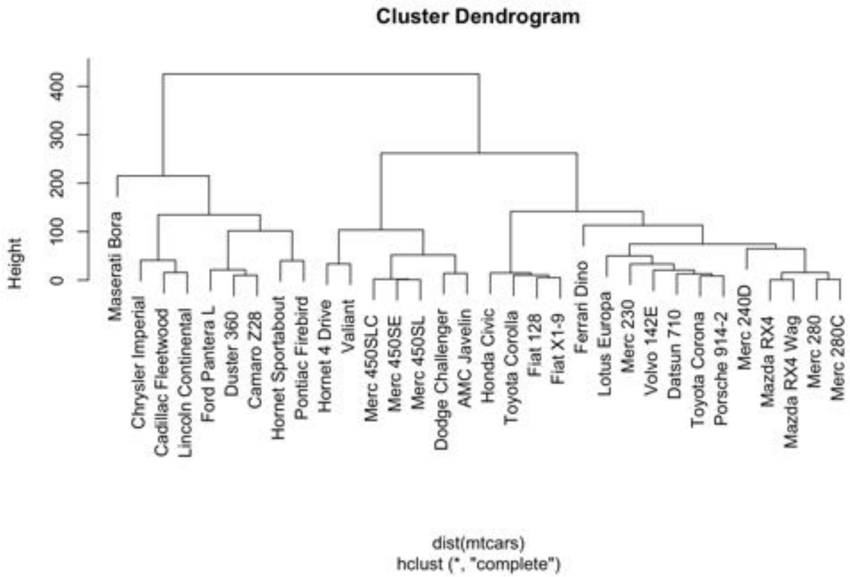
* Principal Component Analysis (PCA) is commonly used for dimensionality reduction. PCA can be thought of as fitting a p-dimensional ellipsoid to the data, where each axis of the ellipsoid represents a principal component. If some axis of the ellipsoid is small, then the variance along that axis is also small.

VISUALIZING CLUSTERING RESULTS (1)

Scatterplot and Dendrogram



Scatterplot of k-means* results color-coded by cluster with cluster centers and cluster bubbles

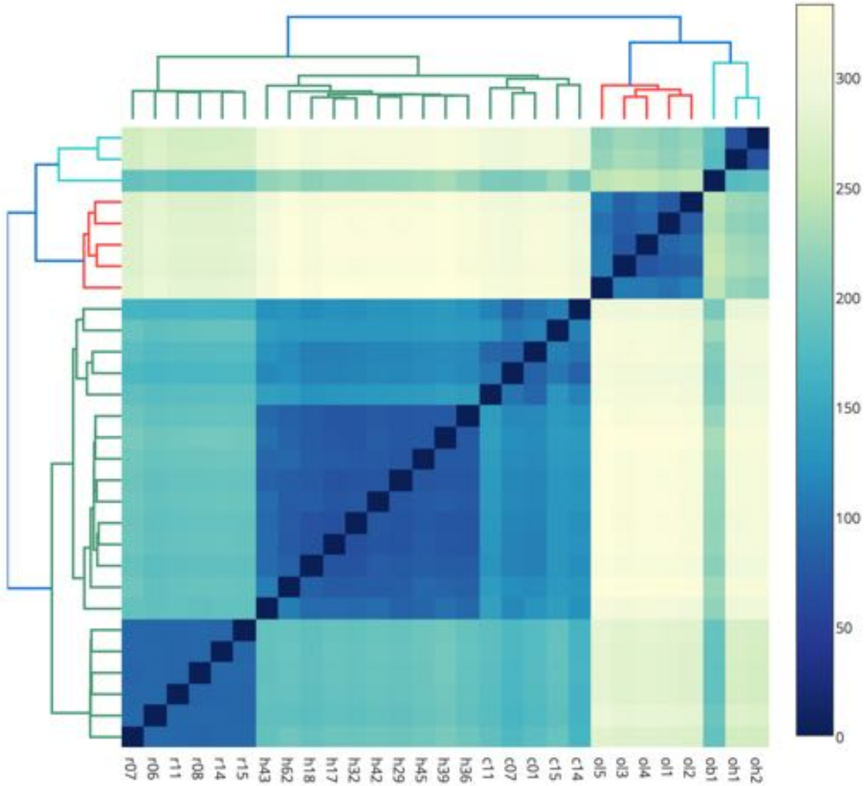


Dendrogram (diagram representing a tree) encoding a value

* k-means clustering is a method of vector quantization, originally from signal processing, that aims to partition n observations into k clusters in which each observation belongs to the cluster with the nearest mean (cluster centers or cluster centroid)

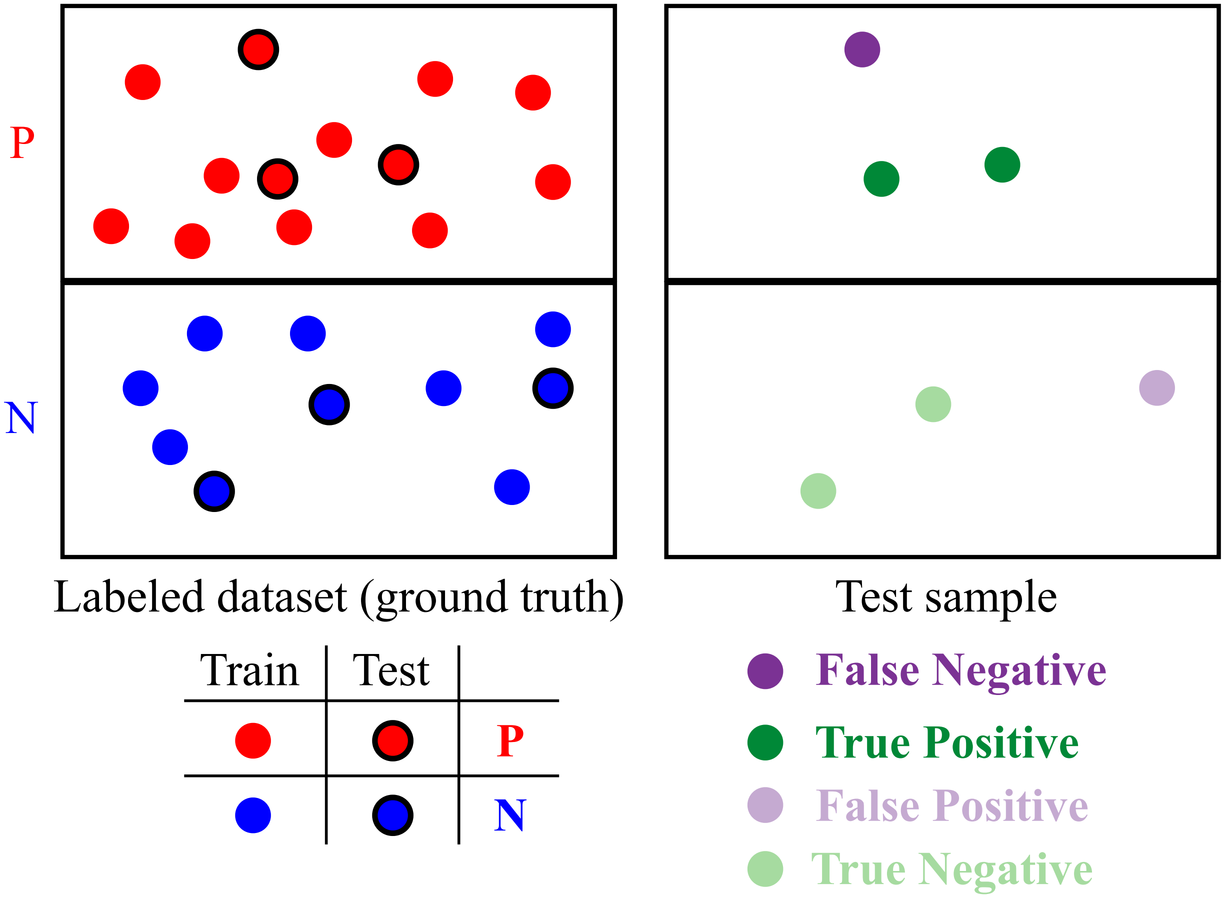
VISUALIZING CLUSTERING RESULTS (2)

Dendrogam and heatmap combo plot



Combination plot of correlation heatmap and dendrograms showing hierarchical information across variables

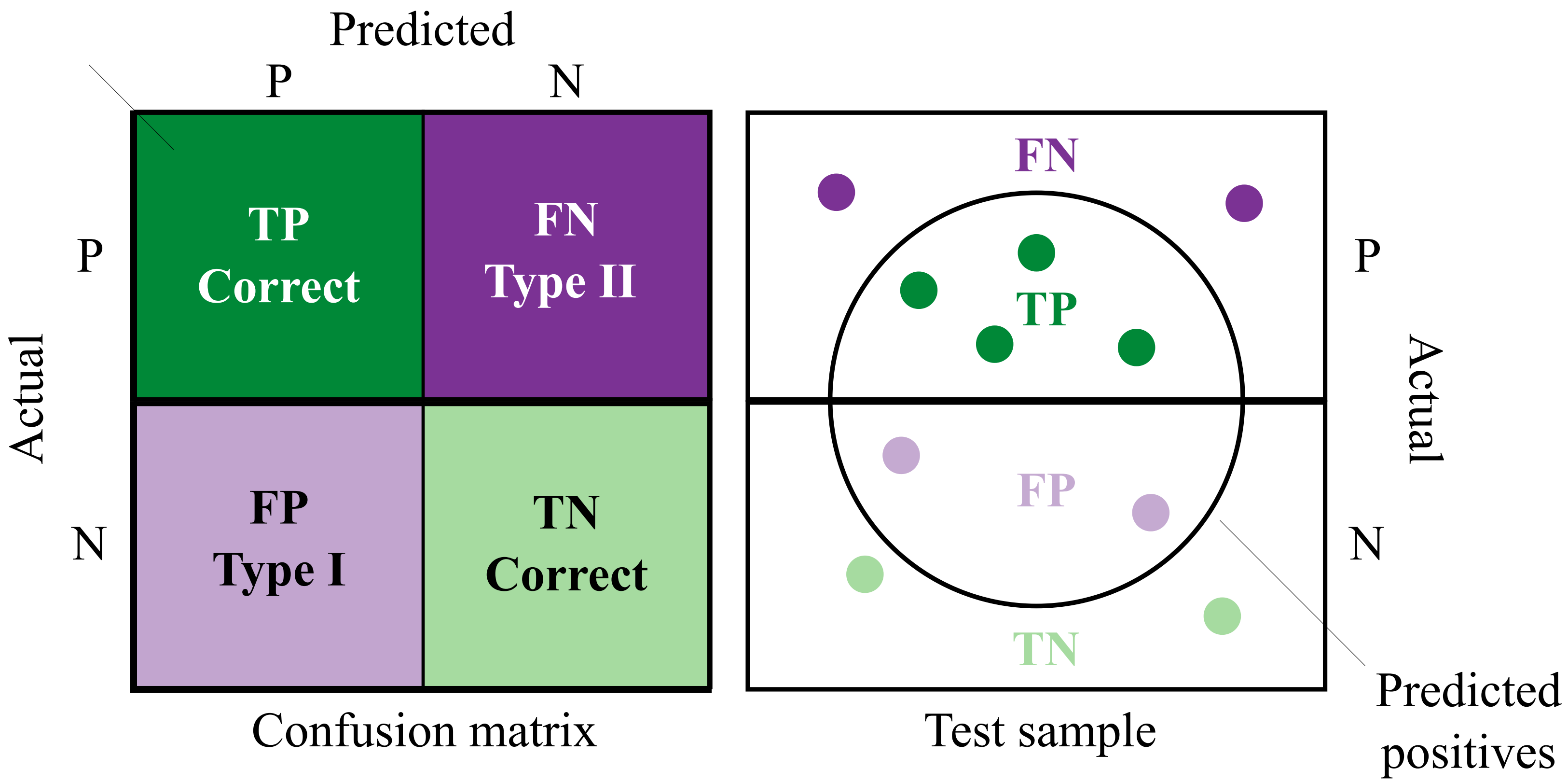
MODEL PERFORMANCE TESTING



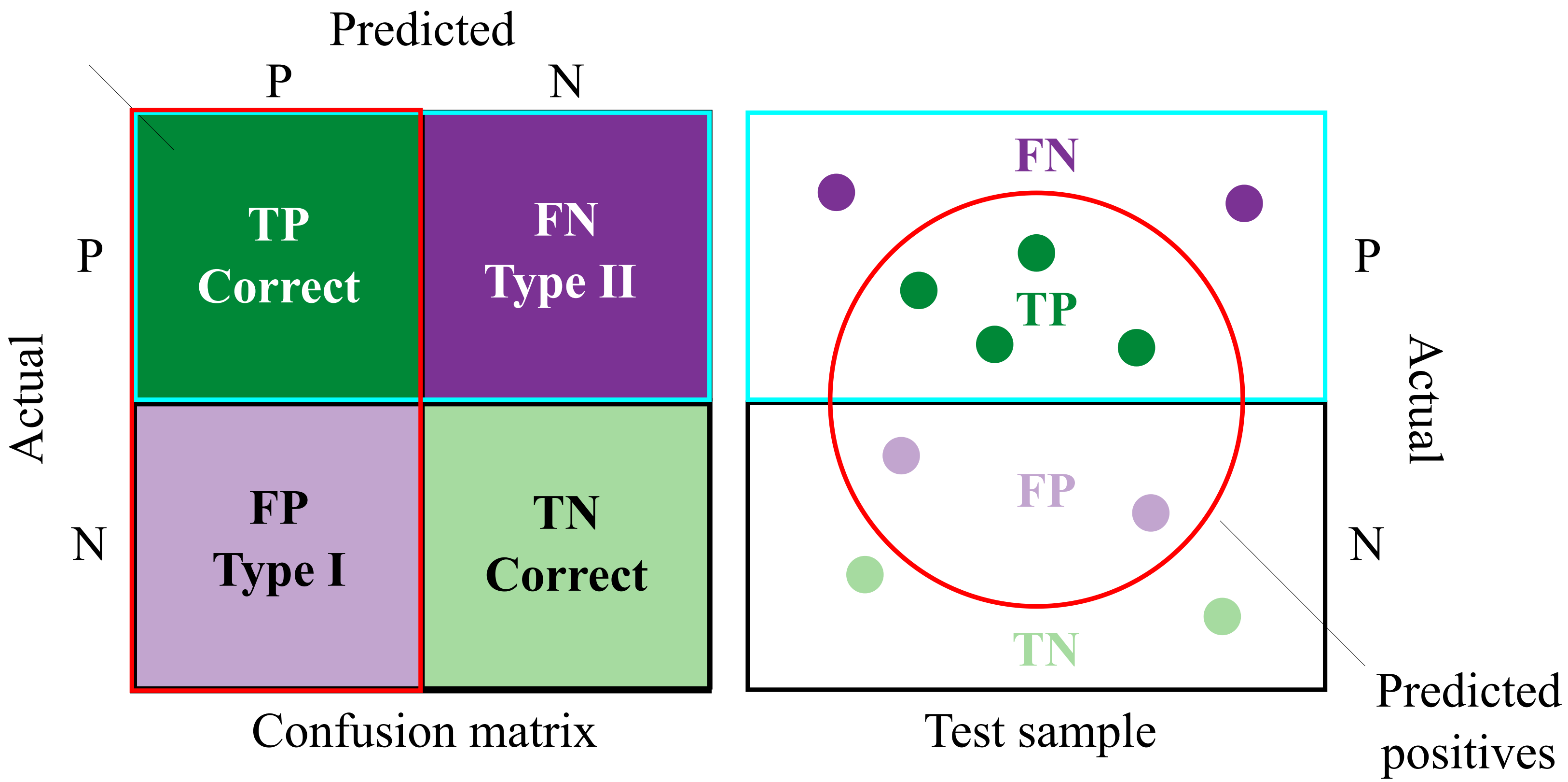
- Frequencies of TP, FP, TN, FN (confusion matrix)
- Precision and Recall rates
- Specificity and Sensitivity rates

Typical testing procedure in supervised learning

CONFUSION MATRIX



PRECISION AND RECALL



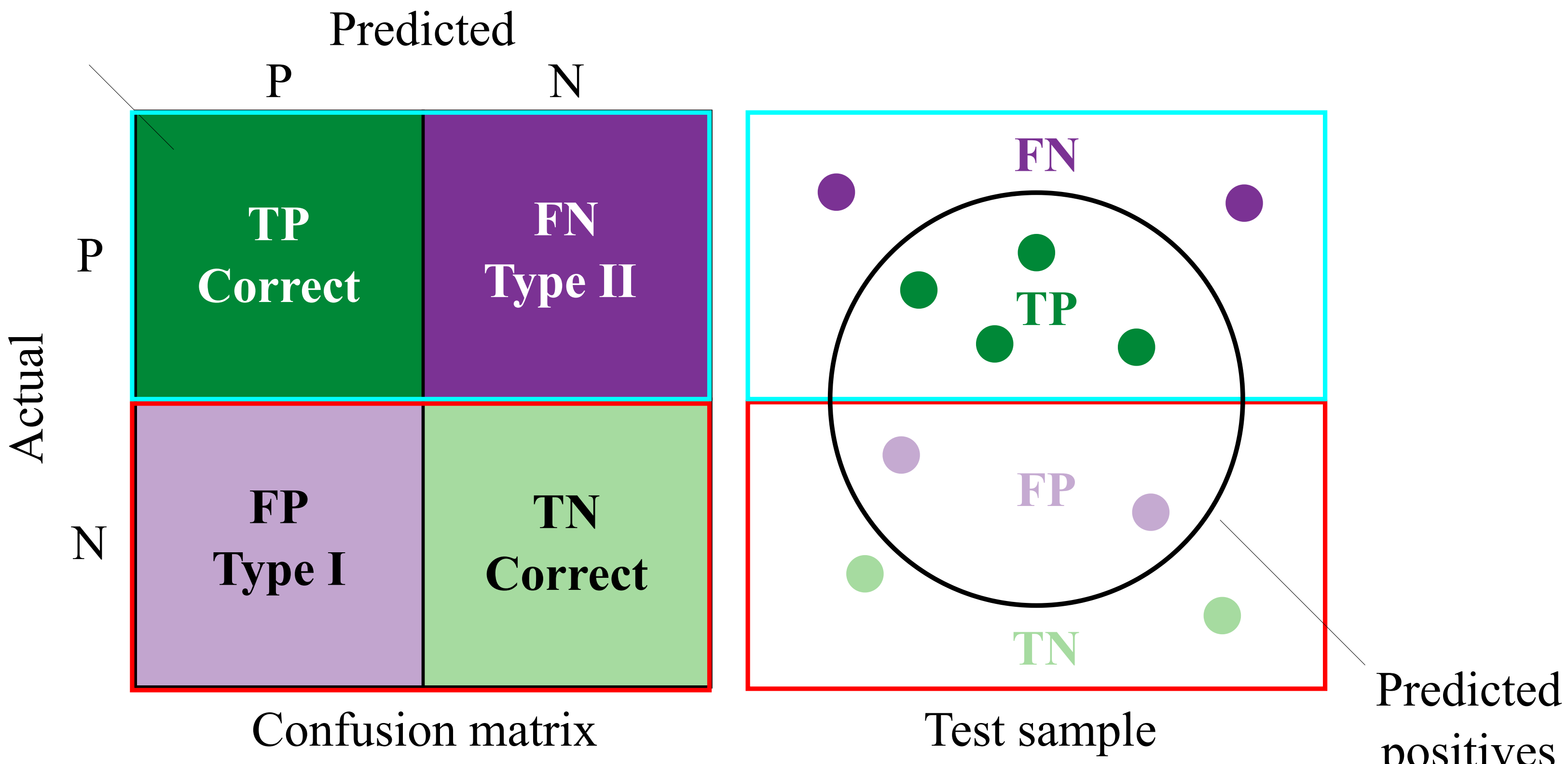
Precision, positive predictive value (PPV)

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

Sensitivity, recall, true positive rate (TPR)

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

SPECIFICITY AND SENSITIVITY



Specificity, selectivity, true negative rate (TNR)

$$\text{Specificity} = \frac{TN}{TN + FP}$$

Sensitivity, recall, true positive rate (TPR)

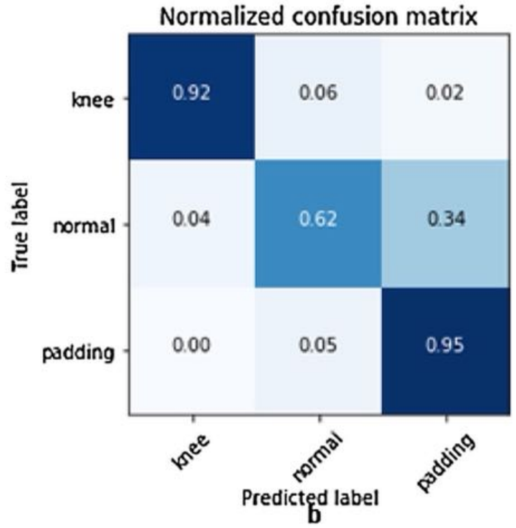
$$\text{Sensitivity} = \frac{TP}{TP + FN}$$

VISUALIZING THE CONFUSION MATRIX

Tables and heatmap

```
# d1:  Int. Derang. (DDWR)  /  Int. Derang. (eDDNR)
 No Yes
188 112

Call:
 randomForest(formula = target, data = df, proximity = TRUE)
        Type of random forest: classification
          Number of trees: 500
No. of variables tried at each split: 11

    OOB estimate of  error rate: 3%
Confusion matrix:
    No  Yes class.error
No  187   1 0.005319149
Yes   8 104 0.071428571
```

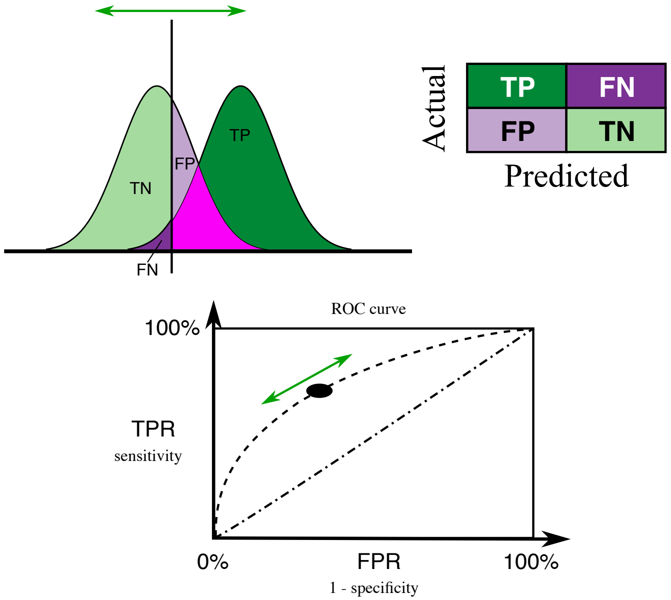
Confusion matrix result in R printed as a table



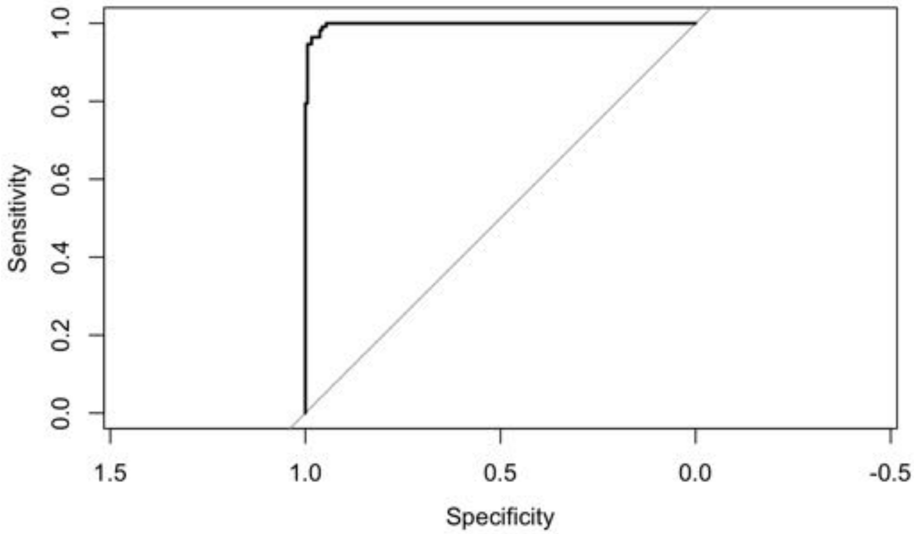
Khokhlova, et al. "Normal and pathological gait classification LSTM model." Artificial intelligence in medicine 94 (2019)

VISUALIZING THE ROC CURVE

Line chart of specificity vs. sensitivity



By Sharpr - Own work, CC BY-SA 3.0, Link



ROC curve of dental Internal Derangement (DDWR/eDDNR) conditions

The Receiver Operator Curve (ROC) is a diagnostic tool for binary classifiers with decision threshold

VISUALIZING PERFORMANCE RATES (1)

Tables to compare classifiers/conditions

CLASSIFIER/COMBINATION	A (%)	P (%)	R (%)	F-M
SVM	84.79	85.43	83.38	0.84
RANDOM FOREST	83.09	83.68	83.09	0.83
K-NN	79.24	80.16	79.24	0.79
DECISION TREE	72.66	72.92	72.66	0.73
NAIVE BAYES	71.02	71.64	71.02	0.70
SKR (AP)	87.41	87.61	87.40	0.87
SK (AP)	87.28	87.49	87.29	0.87
SKR (MV)	85.62	85.79	85.61	0.86
DSK (AP)	85.37	85.61	85.37	0.85
DSK (MV)	85.29	85.51	85.30	0.85

Performance of single classifier and multiple classifiers combination. A: Accuracy, P: Precision, R: Recall, F-M: F-measure, AP: Average of Probabilities, MV: Majority Voting, S: SVM, k: k-NN, D: Decision Tree, R: Random Forest.

TASK	A (%)	P (%)	R (%)	F-M
COUNT	84.79 (93.95/71.23)	85.43	83.38	0.84
TRAY	82.04 (94.44/53.63)	82.19	90.00	0.85
WALK	81.04 (96.05/48.99)	81.63	87.75	0.83

SVM performance for various features. Accuracy is reported with the format as average accuracy (best accuracy/worst accuracy) across 14 subjects. A: Accuracy, P: Precision, R: Recall and F-M: F-measure. ALL: Gait, Angle, and Graph.

FEATURE	A (%)	P (%)	R (%)	F-M
GAIT	63.58 (88.71/39.53)	57.26	55.40	0.51
ANGLE	75.30 (92.22/53.58)	75.01	74.20	0.74
GRAPH	82.41 (95.68/69.63)	83.04	81.93	0.82
ALL	84.79 (93.95/71.23)	85.43	83.38	0.84
PCA	84.66 (95.32/71.99)	85.30	84.44	0.85

SVM performance for various features. Accuracy is reported with the format as average accuracy (best accuracy/worst accuracy) across 14 subjects. A: Accuracy, P: Precision, R: Recall and F-M: F-measure. ALL: Gait, Angle, and Graph.

Kao, J.Y., Nguyen, M., Nocera, L., Shahabi, C., Ortega, A., Winstein, C., Sorkhoh, I., Chung, Y.C., Chen, Y.A. and Bacon, H., 2016, October. Validation of automated mobility assessment using a single 3d sensor. In European Conference on Computer Vision (pp. 162-177). Springer, Cham.

VISUALIZING PERFORMANCE RATES (2)

Bar charts to compare classifiers/conditions

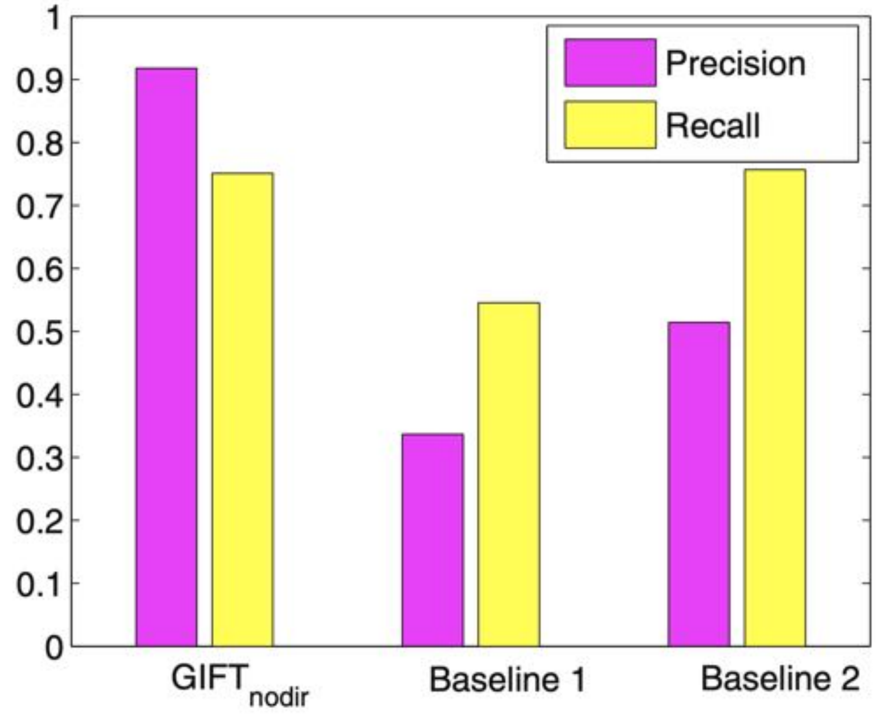


Fig. 19 Average precision and recall after tracking 20 targets

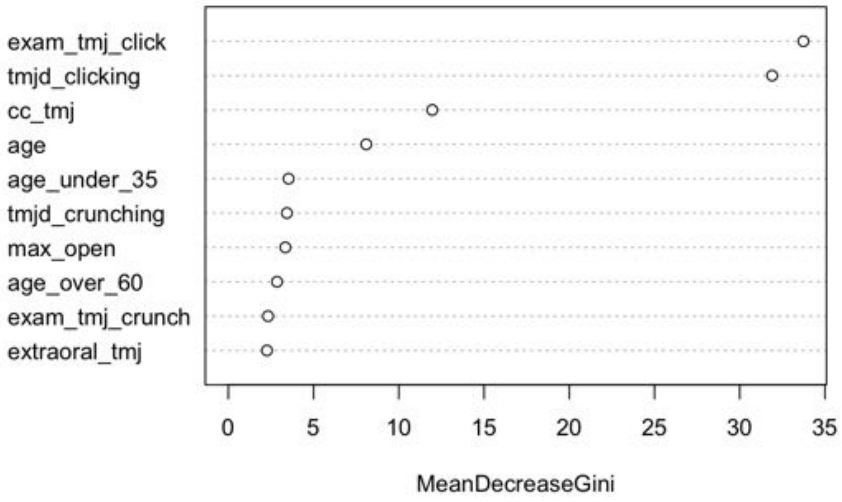
Cai, Y., Lu, Y., Kim, S.H., Nocera, L. and Shahabi, C., 2015, June. Gift: A geospatial image and video filtering tool for computer vision applications with geo-tagged mobile videos. In 2015 IEEE International Conference on Multimedia & Expo Workshops (ICMEW) (pp. 1-6). IEEE.

VISUALIZING FEATURE IMPORTANCE

Table and Dot plot

```
# d1:  Int. Derang. (DDWR)  /  Int. Derang. (eDDNR)
 No Yes
188 112

Call:
 randomForest(formula = target, data = df, proximity = TRUE)
         Type of random forest: classification
           Number of trees: 500
No. of variables tried at each split: 11

    OOB estimate of  error rate: 3%
Confusion matrix:
    No  Yes class.error
No   187    1 0.005319149
Yes    8  104 0.071428571

Top 10 variables
   No    Yes
1  0.990 0.010
2  0.988 0.012
3  0.992 0.008
4  0.108 0.892
5  0.970 0.030
6  0.990 0.010
7  0.962 0.038
8  0.040 0.960
9  0.986 0.014
10 0.042 0.958
Setting levels: control = No, case = Yes
Setting direction: controls < cases
Area under the curve: 0.9974
```

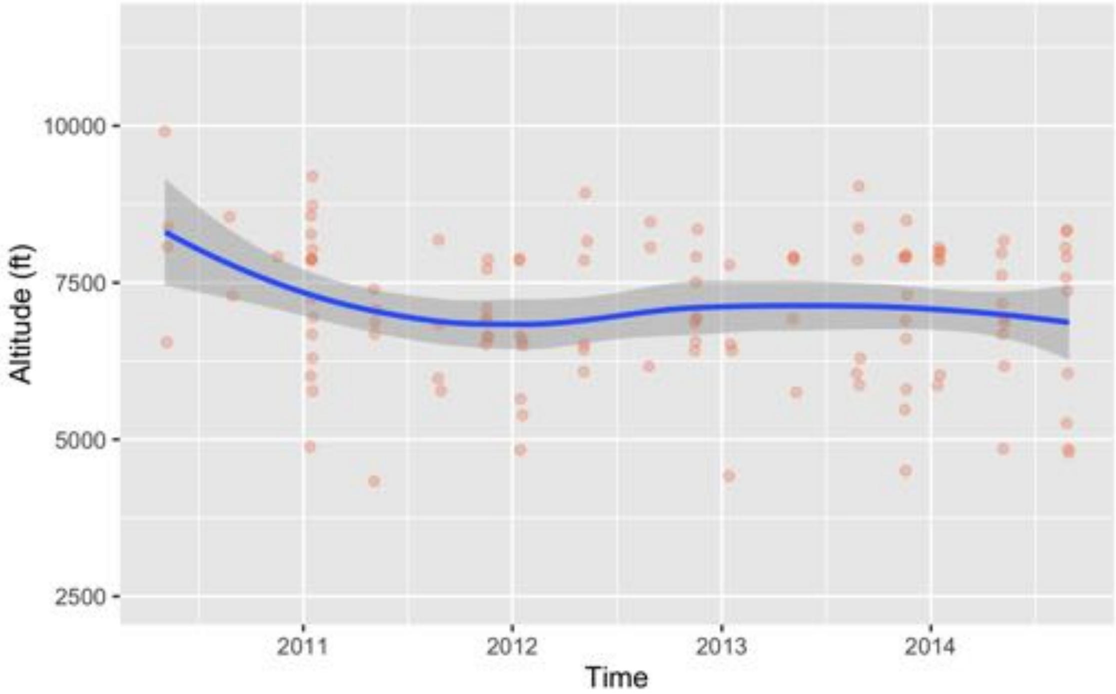
Classification results showing confidence of top 10 variables



Dot plot of mean decrease Gini

VISUALIZING REGRESSION MODELS

Line chart with Ribbon

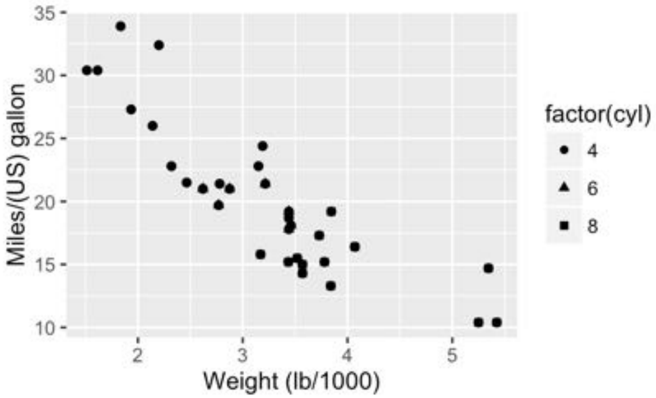


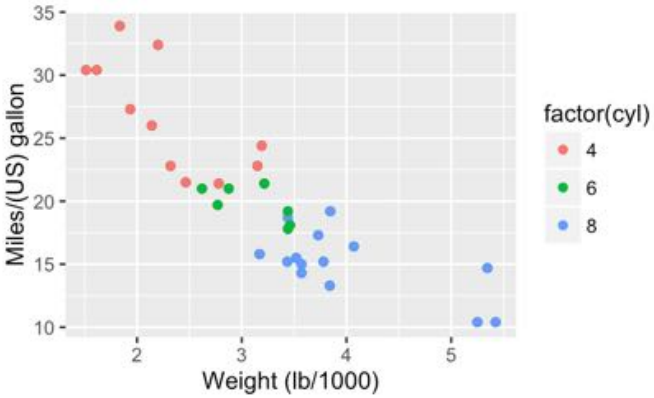
Smooth regression line with 0.95 confidence interval*

*95% confidence interval: interval of values for which a hypothesis test to the level of 5% cannot be rejected ≡ interval has a probability of 95% to contain the true value

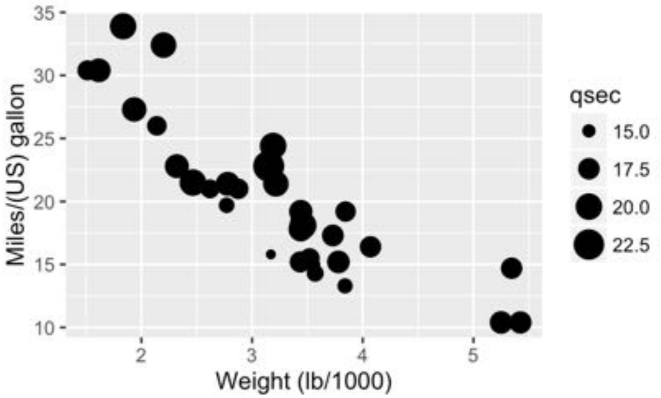
DESIGN: CHOOSE ENCODINGS WISELY

Color & shape work well with categorical variables



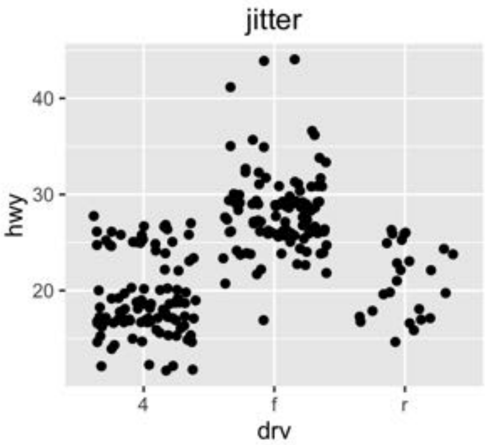


Size works well with continuous variables

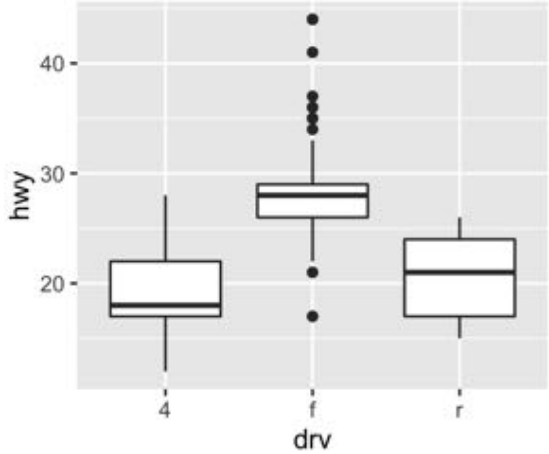


DESIGN: DEAL WITH OVERPLOTTING

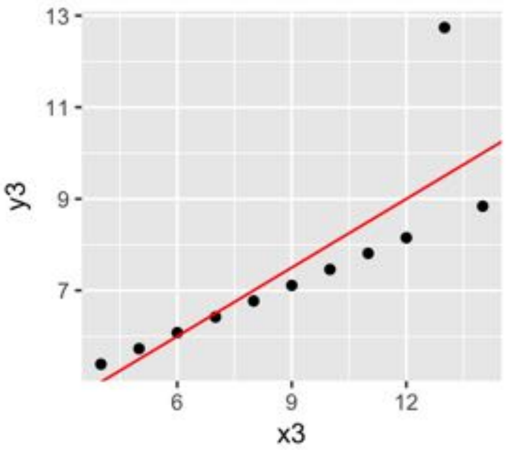
Transparency, outline shape



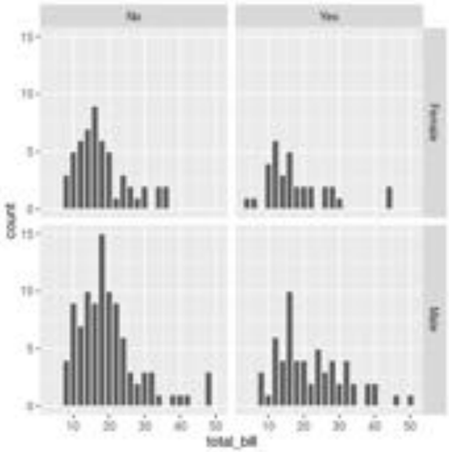
Add jitter



Summarize the data



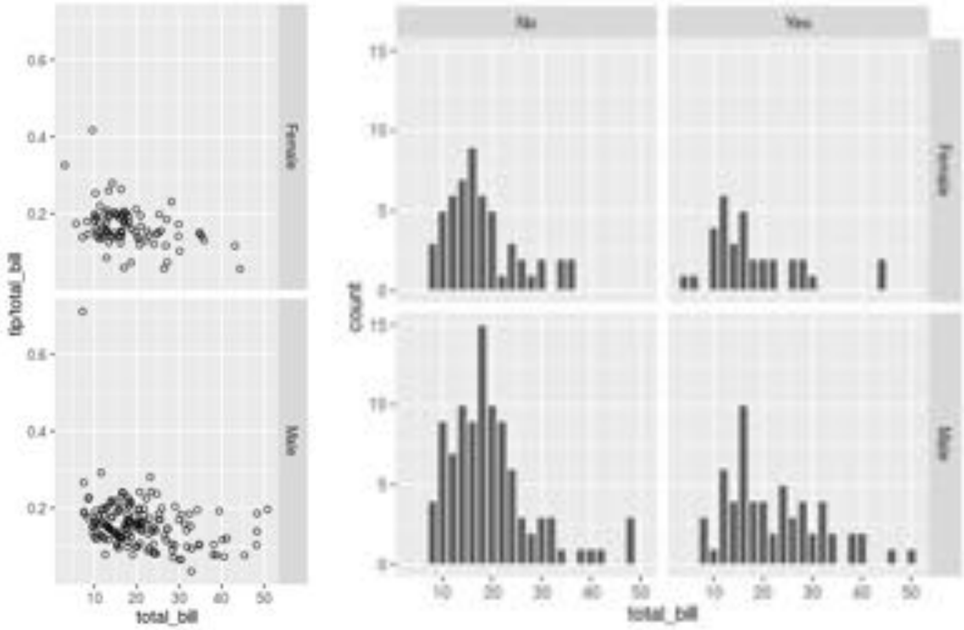
Add information



Split the data

DESIGN: SERIES ARE BETTER THAN COMPLEX PLOTS

Faceting/conditioning/latticing/trellising/small multiples



OUTLINE

- Basics of statistics and modeling
- Statistical graphics
- Tools

VISUALIZATION TOOLS

Ease-of-Use

Chart Typologies
Excel, Google Sheets, Matplotlib, Seaborn

Visual Analysis Grammars
VizQL, Tableau, ggplot2, plotnine, Altair

Visualization Grammars
Protovis, D3, Vega, Vega-Lite

Component Architectures
Prefuse, Flare, Improvise, VTK

Graphics applications
Processing, P5.js, WebGL, three.js, OpenGL

Expressiveness

already covered — covered this week — will discuss later

Adapted from [Heer 2014]

WORKING IN NOTEBOOKS

Data is in dataframe format:

- Same length columns
- Columns → variables
- Rows → observations
- Strings stored as pointers (R factors)

```
> df <- sample_n(mpg, 36)
> df$manufacturer <- factor(df$manufacturer)
> df
# A tibble: 36 x 11
  manufacturer model              displ  year   cyl trans      drv     cty   hwy fl    class
  <fct>        <chr>              <dbl> <int> <int> <chr>      <chr> <int> <int> <chr> <chr>
  1 toyota       camry                2.4  2008     4 auto(l5)   f        21    31 r     midsize
  2 toyota       camry solara         2.4  2008     4 manual(m5) f        21    31 r     compact
  3 dodge        dakota pickup 4wd    4.7  2008     8 auto(l5)   4         9    12 e     pickup
  4 chevrolet    corvette             5.7  1999     8 auto(l4)   r        15    23 p     2seater
  5 audi         a4                   1.8  1999     4 manual(m5) f        21    29 p     compact
  6 jeep         grand cherokee 4wd   4.7  1999     8 auto(l4)   4        14    17 r     suv
  7 hyundai      tiburon              2    1999     4 manual(m5) f        19    29 r     subcompact
  8 dodge        dakota pickup 4wd    3.9  1999     6 manual(m5) 4        14    17 r     pickup
  9 toyota       camry solara         3    1999     6 auto(l4)   f        18    26 r     compact
10 ford         expedition 2wd       4.6  1999     8 auto(l4)   r        11    17 r     suv
# ... with 26 more rows
> summary(df$manufacturer)
audi  chevrolet      dodge       ford      honda    hyundai       jeep land rover
   3          2          5          5          2          2          2          1
nissan    pontiac     subaru     toyota volkswagen
   2          1          1          7          3
```

PROPER DATAFRAME FORMAT?

	Granite	Limestone	Sandstone
Trad	36	0	52
Sport	76	8	41
Bouldering	102	0	13

Counts of locations by rock type and type of rock climbing

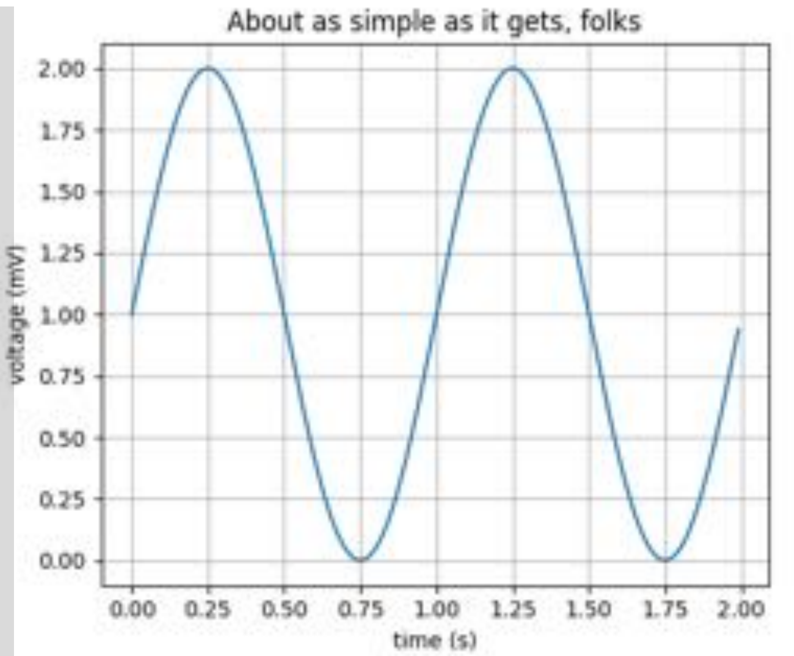
PROPER DATAFRAME FORMAT

rock	type	count
Granite	Trad	36
Granite	Sport	76
Granite	Bouldering	102
Limestone	Trad	0
Limestone	Sport	8
Limestone	Bouldering	0
Sandstone	Trad	52
Sandstone	Sport	41
Sandstone	Bouldering	13

MATPLOTLIB

- http://matplotlib.org and gallery
- Chart typology
- Originally emulating the MATLAB® graphics commands
- Imperative (functional) programming

```python
import matplotlib.pyplot as plt
import numpy as np

T = np.arange(0.0, 2.0, 0.01)
S = 1 + np.sin(2*np.pi*t)

plt.plot(T, S)
plt.xlabel('time (s)')
plt.ylabel('voltage (mV)')
plt.title('About as simple as it gets, folks')
plt.grid(True)

plt.show()
```



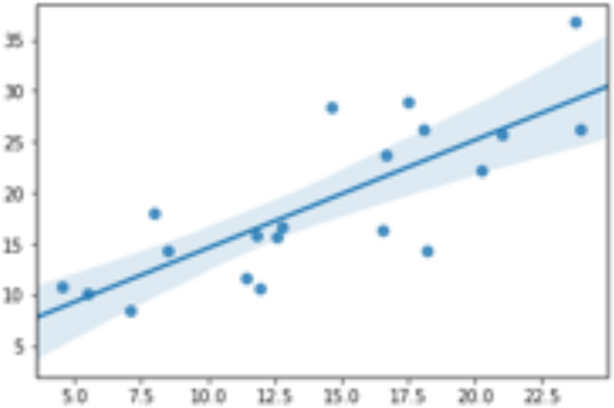
SEABORN

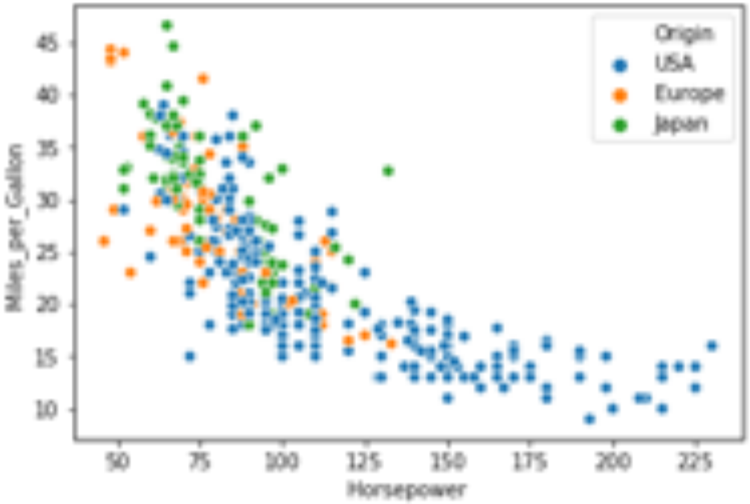
- https://seaborn.pydata.org and gallery
- Chart typology
- High-level interface for statistical graphics based on Matplotlib
- Imperative (functional) programming
- Support for Pandas dataframes

```
 Acceleration Cylinders  Displacement Horsepower Miles_per_Gallon Name                      Origin Weight_in_lbs Year
0 12.0        8          307.0        130.0      18.0             chevrolet chevelle malibu USA    3504          1970-01-01
1 11.5        8          350.0        165.0      15.0             buick skylark 320         USA    3693          1970-01-01
2 11.0        8          318.0        150.0      18.0             plymouth satellite        USA    3436          1970-01-01
3 12.0        8          304.0        150.0      16.0             amc rebel sst             USA    3433          1970-01-01
4 10.5        8          302.0        140.0      17.0             ford torino               USA    3449          1970-01-01
 ...
```

```
import numpy as np
import seaborn as sns

x = 5 + np.arange(20) +
    np.random.randn(20)
y = 10 + np.arange(20) +
  5 * np.random.randn(20)

sns.regplot(x, y)
```

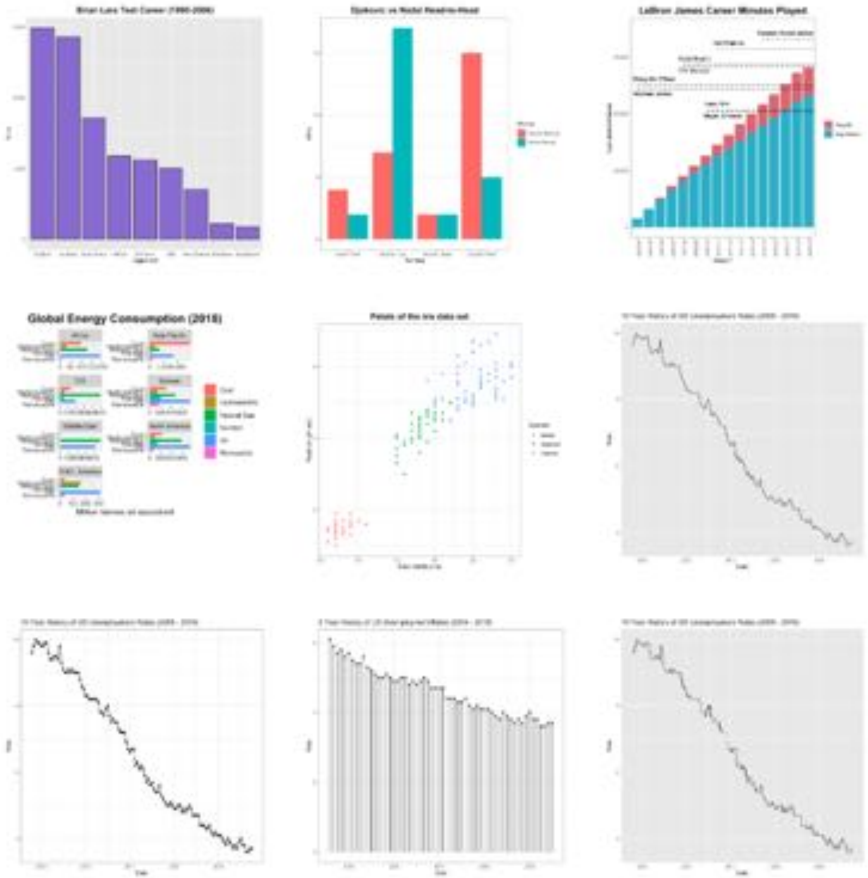
```
import seaborn as sns
from vega_datasets import data

cars = data.cars()
sns.scatterplot(
  x='Horsepower',
  y='Miles_per_Gallon',
  hue='Origin',
  data=cars);
```





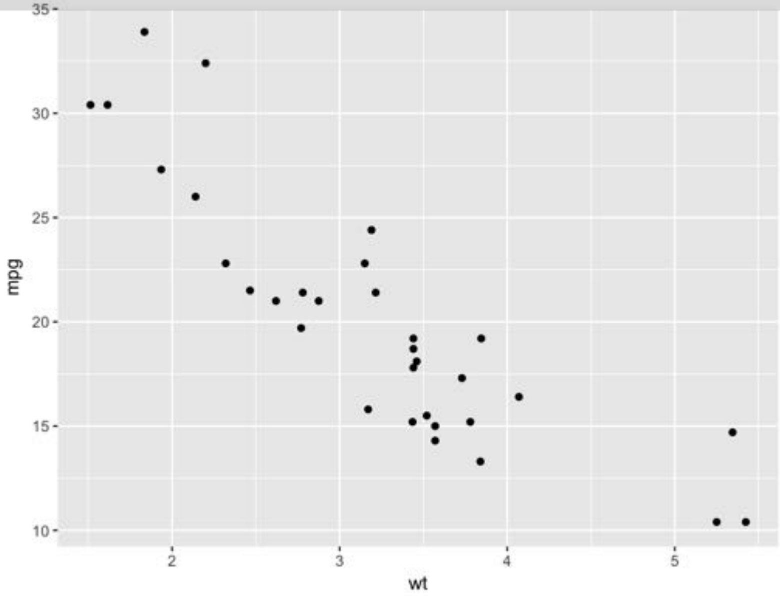
GGPLOT2

- ggplot2 R package and ggg gallery
- Visual Analysis Grammar
- Support for R dataframes




```
mpg cyl disp  hp drat    wt  qsec vs am gear carb
Mazda RX4      21.0 6 160.0 110 3.90 2.620 16.46  0  1    4    4
Mazda RX4 Wag  21.0 6 160.0 110 3.90 2.875 17.02  0  1    4    4
Datsun 710     22.8 4 108.0  93 3.85 2.320 18.61  1  1    4    1
...
```

```
#ggplot(Data, Mapping) + Geom
ggplot(mtcars, aes(x=wt, y=mpg)) + geom_point()
```

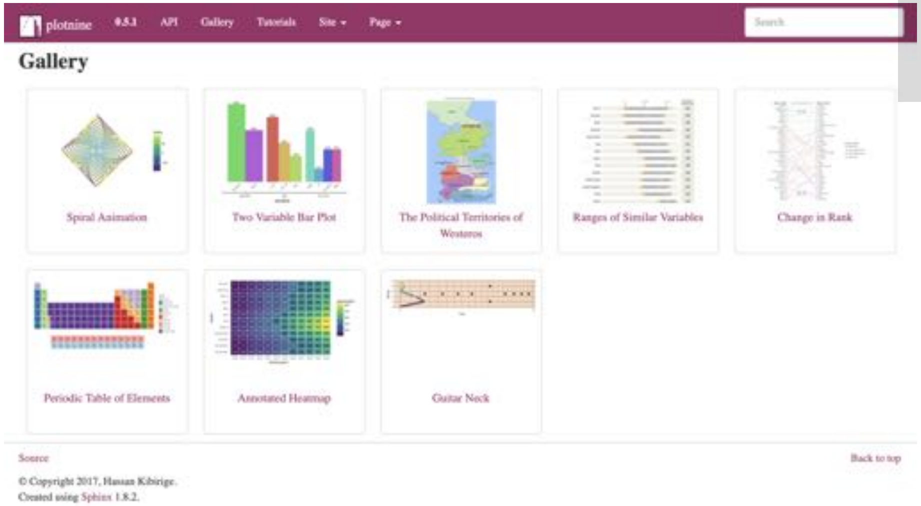


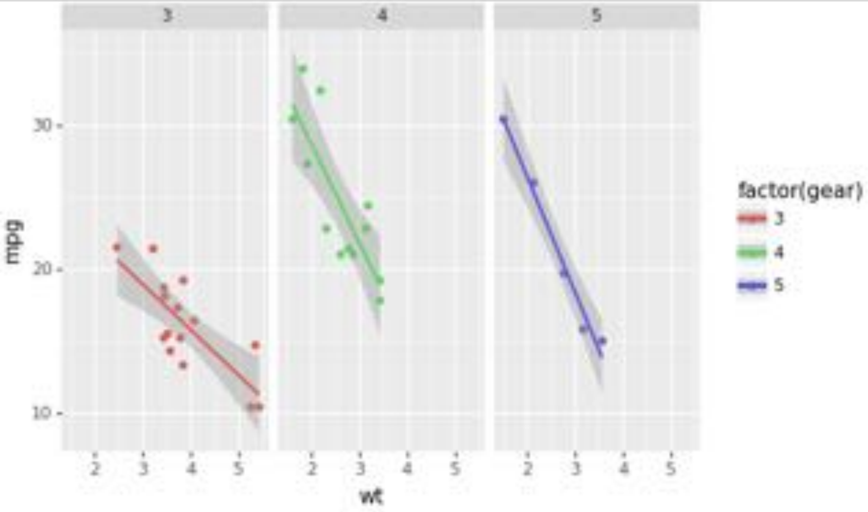
PLOTNINE

- Plotnine website and gallery
- Visual Analysis Grammar
- Based on ggplot2 for Python
- Support for Pandas dataframes

```
                mpg cyl disp  hp drat    wt  qsec vs am gear carb
Mazda RX4      21.0 6 160.0 110 3.90 2.620 16.46  0  1    4    4
Mazda RX4 Wag  21.0 6 160.0 110 3.90 2.875 17.02  0  1    4    4
Datsun 710     22.8 4 108.0  93 3.85 2.320 18.61  1  1    4    1
...
```

```
(ggplot(mtcars, aes('wt', 'mpg', color='factor(gear)'))
+ geom_point()
+ stat_smooth(method='lm')
+ facet_wrap('~gear'))
```

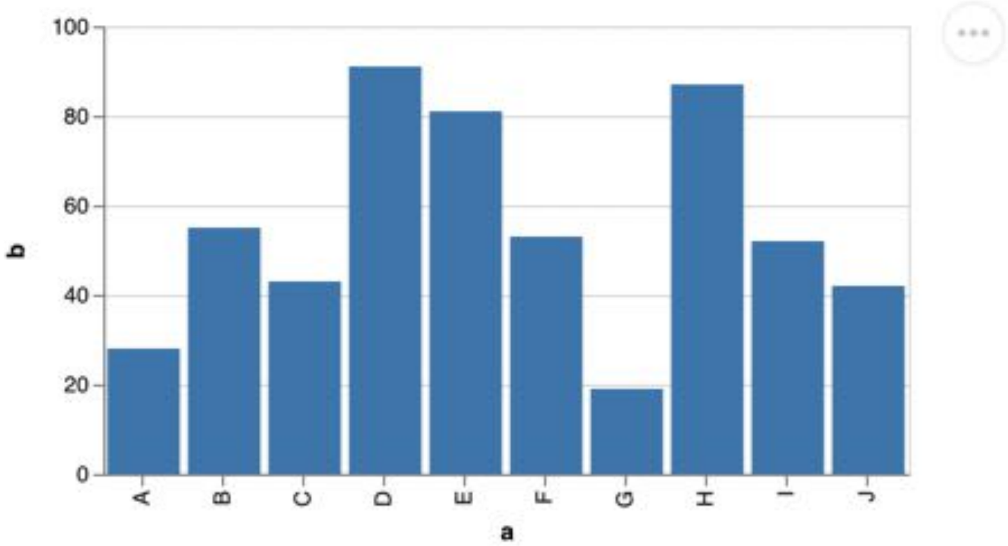




ALTAIR

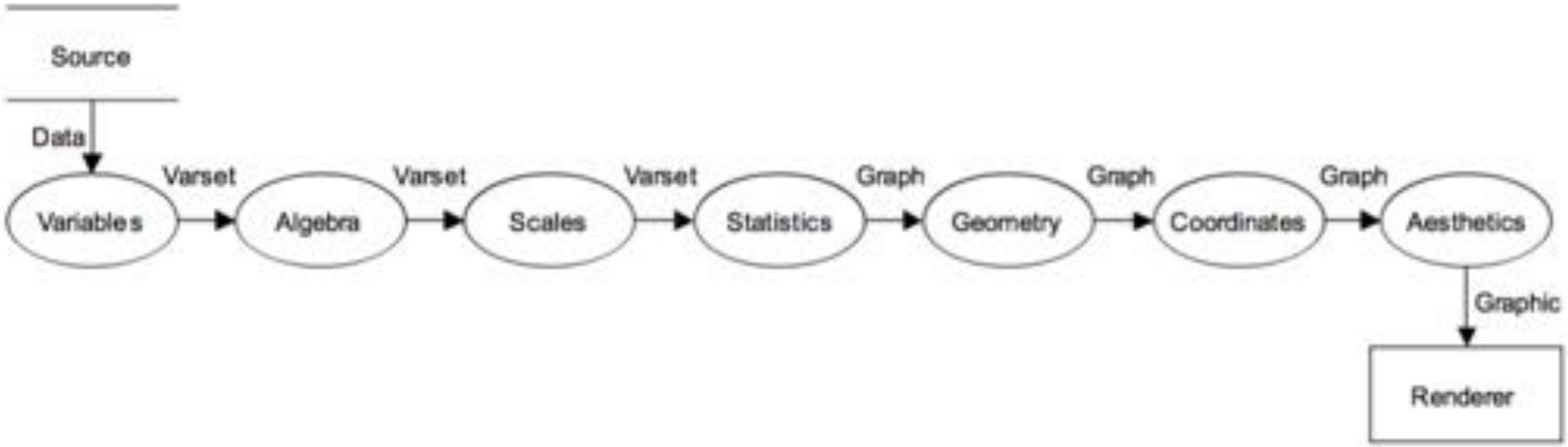
- Altair website and gallery
- Visual Analysis Grammar
- Declarative synthax
- Statistical visualization library
- Based on Vega and Vega-Lite
- Support for Pandas dataframes

```
import altair as alt

# load a simple dataset as a pandas DataFrame
from vega_datasets import data
cars = data.cars()

alt.Chart(cars).mark_point().encode(
  x='Horsepower',
  y='Miles_per_Gallon',
  color='Origin',
).interactive()
```



GRAMMAR OF GRAPHICS*

Graphic defined by a **grammar** of **components**



1. DATA: a set of data operations that create variables from datasets,
2. TRANS: variable transformations, e.g., rank,
3. SCALE: scale transformations, e.g., log,
4. COORD: a coordinate system, e.g., polar,
5. ELEMENT: graphs, e.g., points, and their aesthetic attributes, e.g., color,
6. GUIDE: one or more guides, e,g., axes, legends.

*Wilkinson, L. (2005), The Grammar of Graphics (2nd ed.). Statistics and Computing, New York: Springer

LAYERED GRAMMAR OF GRAPHICS* [WICKHAM 2010]

Defaults Data Mapping**	A default dataset and set of mappings from variables to aesthetics
Layer Data Mapping Geom Stat Position	One or more layers, each composed of a geometric object, a statistical transformation, a position adjustment, and optionally, a dataset and aesthetic mappings
- Coord - Facet	A coordinate system The facetting specification

A theme controls the finer points of display, like the font size and background color

* implemented in **ggplot2**
** Mapping of visual properties to data columns is referred to as an **aesthetic mapping**

MINIMAL GGPLOT2 PLOT

3 components required in every ggplot2 plot: data, aesthetic mapping, Geom

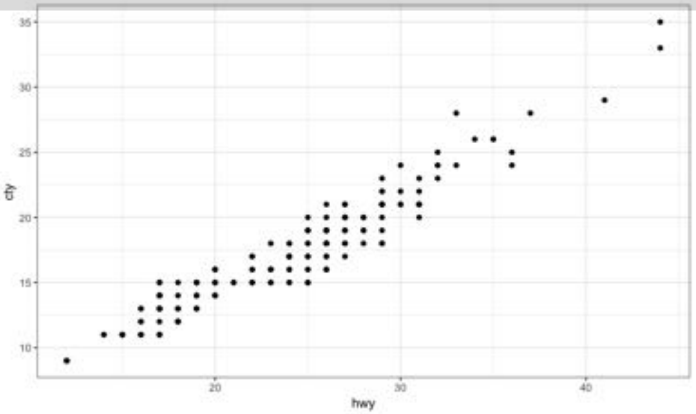
Defaults
- Data
- Mapping

Layer
- Data
- Mapping
- Geom
- Stat
- Position

Scale

Coord
Facet

```
ggplot(data=mpg, aes(x=hwy, y=cty)) + geom_point() #Defaults
ggplot(mpg, aes(hwy, cty)) + geom_point() #positional args
ggplot(mpg) + geom_point(aes(hwy, cty))  #Mapping in layer

# Same using a variable
p <- ggplot(mpg, aes(hwy, cty))  #set Defaults
p + geom_point()  #add Layer with Geom
```



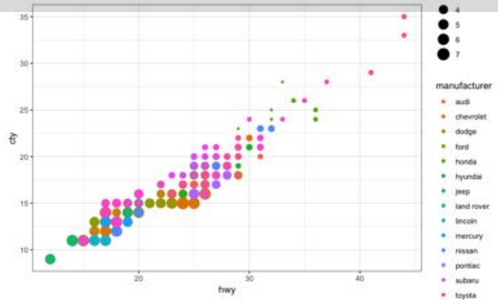
AESTHETIC MAPPINGS

- Geom defines the marks
- Aesthetic mappings allow to map data variables to axes and channels (mark attributes such as position, shape, size, or color)

aes() is used to reference variables in the dataframe

```
ggplot(data=mtcars, aes(x=mpg, y=wt)) + geom_point() #Defaults

# mtcars dataset:
               mpg  cyl  disp  hp drat    wt  qsec vs am gear carb
Mazda RX4     21.0    6 160.0 110 3.90 2.620 16.46  0  1    4    4
Mazda RX4 Wag 21.0    6 160.0 110 3.90 2.875 17.02  0  1    4    4
Datsun 710    22.8    4 108.0  93 3.85 2.320 18.61  1  1    4    1

aes(x = mpg, y = wt)
#> Aesthetic mapping:
#> * `x` -> `mpg`
#> * `y` -> `wt`

# You can also map aesthetics to functions of variables
aes(x = mpg ^ 2, y = wt / cyl)
#> Aesthetic mapping:
#> * `x` -> `mpg^2`
#> * `y` -> `wt/cyl`

# Or to constants
aes(x = 1, colour = "smooth")
#> Aesthetic mapping:
#> * `x`      -> 1
#> * `colour` -> "smooth"
```

```
# Named arguments
ggplot(mpg, aes(x=hwy, y=cty, color=manufacturer, size=displ)) + geom_point()

# Positional & named
ggplot(mpg, aes(hwy, cty, color=manufacturer, size=displ)) + geom_point()

# Using abbreviation for color
ggplot(mpg, aes(hwy, cty, col=manufacturer, size=displ)) + geom_point()

# Using english spelling for color
ggplot(mpg, aes(hwy, cty, colour=manufacturer, size=displ)) + geom_point()

# Specifying aesthetic mappings in geom layer
ggplot(mpg, aes(hwy, cty)) + geom_point(aes(color=manufacturer, size=displ))

  # Wrong: color and size are not mapped with aes
ggplot(mpg, aes(hwy, cty), color=manufacturer, size=displ) + geom_point()
ggplot(mpg, aes(hwy, cty)) + geom_point(color=manufacturer, size=displ)
```



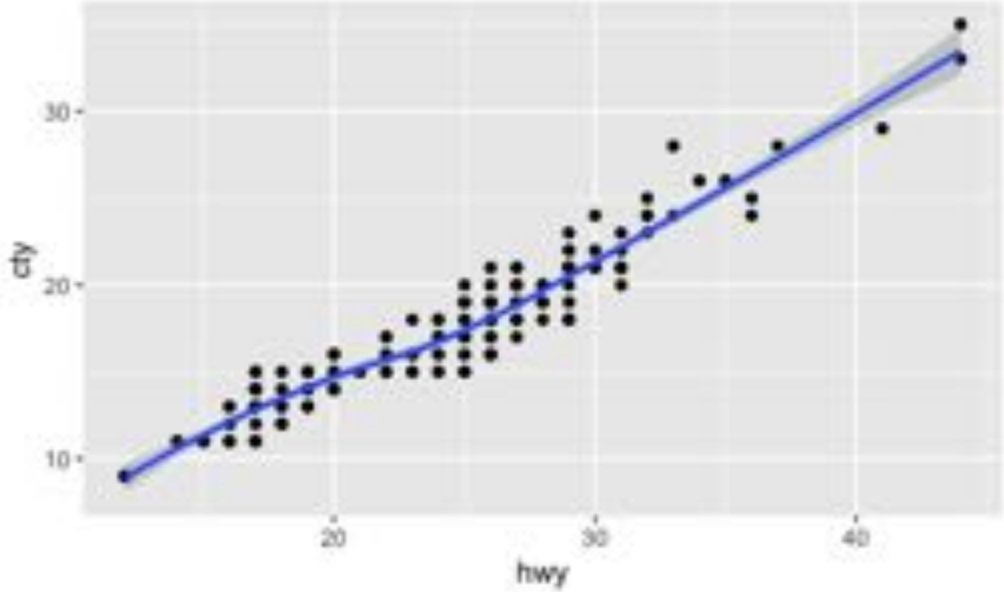
ADDING LAYERS

Defaults
  Data
  Mapping

Layer
  Data
  Mapping
  Geom
  Stat
  Position
Scale

Coord
Facet

```
> ggplot(mpg, aes(hwy, cty)) +  #Defaults
  geom_point() +  #add Geom point Layer
  geom_smooth()  #add Geom smooth Layer (regression)
```



BASIC NAMED PLOTS

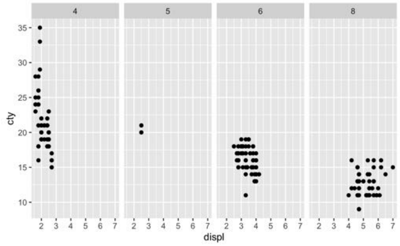
All understand x, y, color and size aesthetics.
Filled geoms also understand fill.

Scatterplot	geom_point()
Text	geom_text()
Bar chart	geom_bar()
Line chart	geom_line()
Area chart	geom_area()
Dot plot	geom_dotplot()
Histogram	geom_histogram()
Frequency polygon	geom_freqpoly()
Box plot	geom_boxplot()
Violin plot	geom_violin()

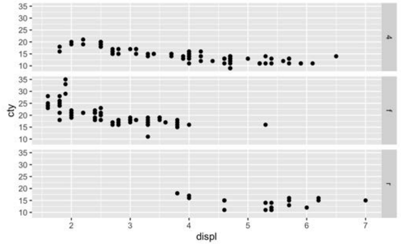
FACETING: `FACET_GRID`

```
p <- ggplot(mpg, aes(displ, cty)) + geom_point()
```

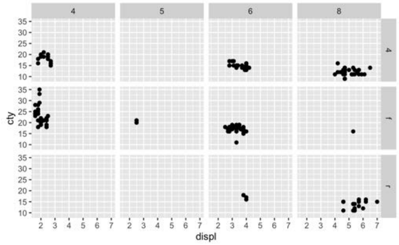
By columns



By rows



By rows & columns




```
# New notation
p + facet_grid(cols = vars(cyl))

# Model notation: no faceting in y
p + facet_grid(. ~ cyl)
```

```
# New notation
p + facet_grid(rows = vars(drv))

# Model notation: no faceting in x
p + facet_grid(drv ~ .)
```

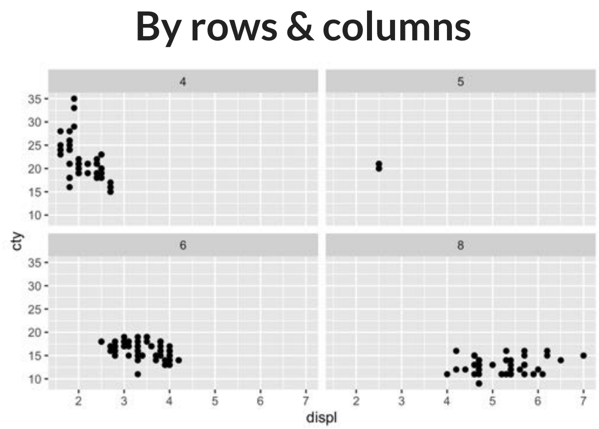
```
# New notation: facet_grid(rows, cols)
p + facet_grid(vars(drv), vars(cyl))

# Model notation: facet_grid(y ~ x)
p + facet_grid(drv ~ cyl)
```

```
# R model formula
y ~ x  # ~ separates the left- and right-hand sides in the model formula
fit <- lm(y ~ x1 + x2 + x3, data=mydata)  #example of multiple linear regression
```

dot in the model formula indicates no faceting in that dimension.

FACETING: FACET_WRAP

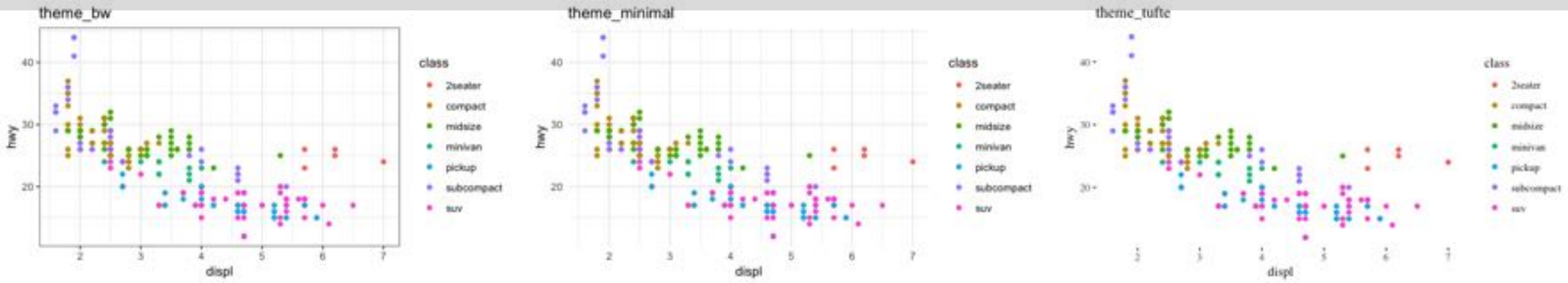
```
p <- ggplot(mpg, aes(displ, cty)) + geom_point()
```




```
# New notation: facet_grid(rows, cols)
p + facet_wrap(facets=vars(lf))

# Model notation: facet_grid(y ~ x)
p + facet_grid(~ lf)
```

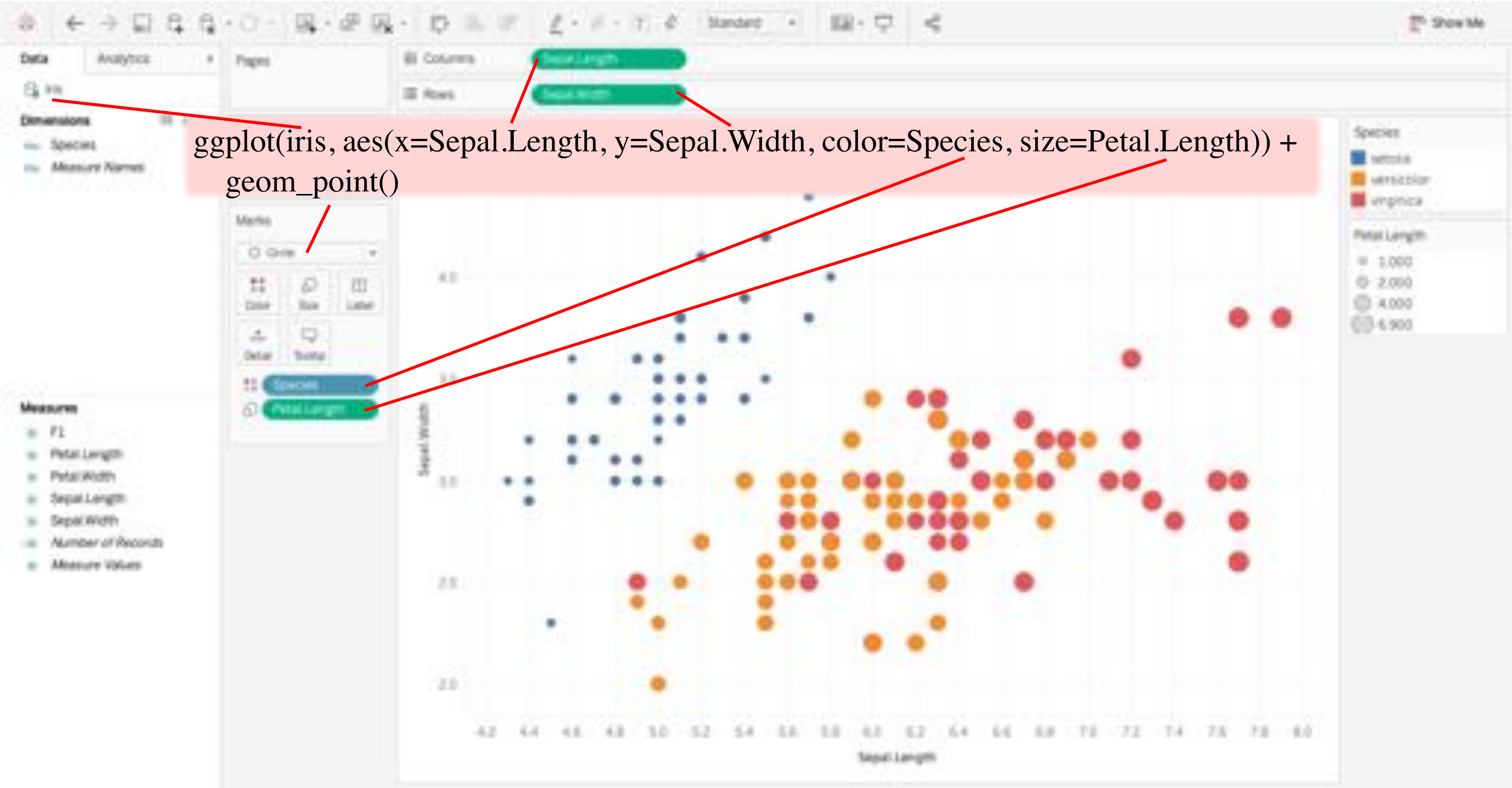
THEMES

```
p <- ggplot(mpg, aes(displ, hwy, color=class)) + geom_point()
p + theme_bw() + ggtitle("theme_bw")
p + theme_minimal() + ggtitle("theme_minimal")

library(ggthemes)  #extra themes
p + theme_tufte() + ggtitle("theme_tufte")

theme_set(theme_bw())  #sets the theme for all subsequent ggplot plots
```



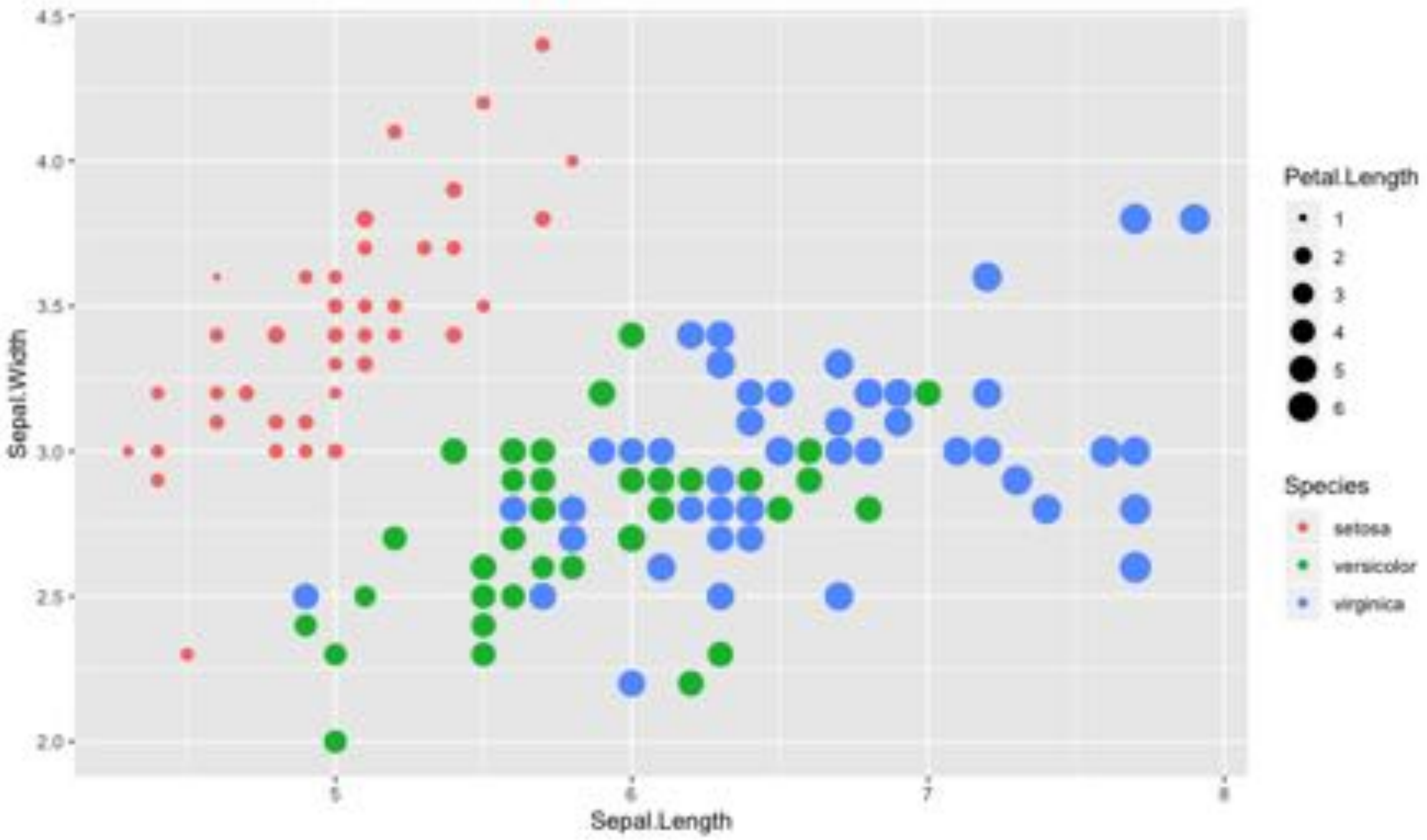
Extra themes in package ggthemes

TABLEAU VS. GGPLOT2



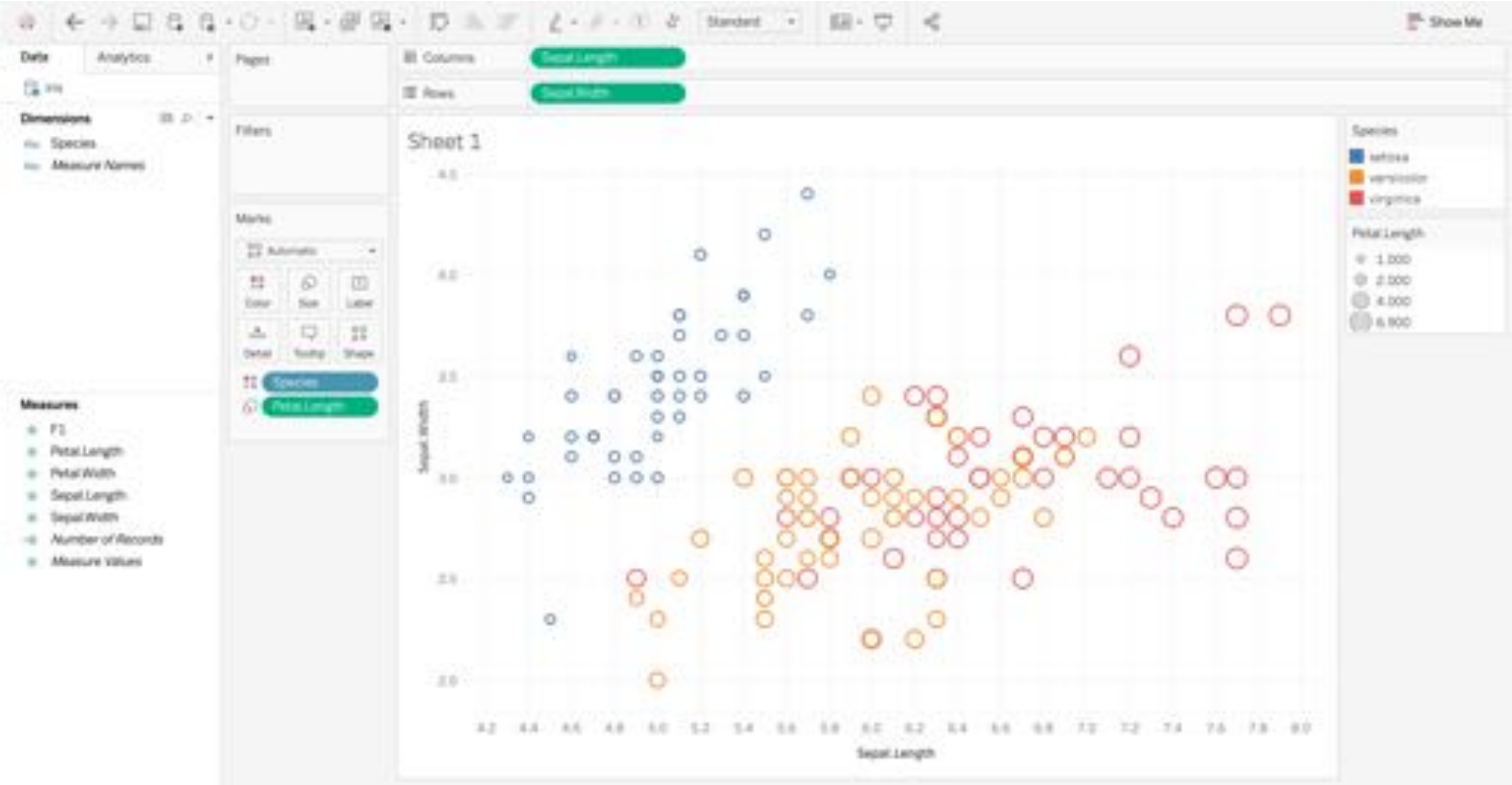
GGPLOT 2 (LAYERED GRAMMAR)

```
ggplot(iris, aes(x=Sepal.Length, y=Sepal.Width, color=Species, size=Petal.Length)) + geom_point()
```



Use `geom_point(shape=1)` to draw circle outline

TABLEAU (VISUAL GRAMMAR)



With data read from CSV:
Dimensions ↔ categorical variables
Measures ↔ numerical variables