Statistical Learning Tutorial

Data

Data is characteristics or information, usually numerical, collected through observation. In a more technical sense, data is a set of values of qualitative or quantitative variables about one or more persons or objects, while a datum (singular of data) is a single value of a single variable.

There are two types of data:

Continuous: Continuous data is data that can take any value. Height, weight, temperature, and length are all examples of continuous data. Some continuous data will change over time; for example, the weight of a baby in its first year or the temperature in a room throughout the day.

Categorical: Categorical variables represent types of data that may be divided into groups. Examples of categorical variables include race, sex, age group, and educational level.

Level of Measurements

Level of measurement or scale of measure is a classification that describes the nature of information within the values assigned to variables. Psychologist Stanley Smith Stevens developed the best-known classification with four levels, or scales, of measurement: nominal, ordinal, interval, and ratio. This framework of distinguishing levels of measurement originated in psychology and is widely criticized by scholars in other disciplines. Other classifications include those by Mosteller and Tukey, and by Chrisman.

Nominal Measurement: A nominal variable is one of the two types of categorical variables and is the simplest among all measurement variables. Some examples of nominal variables include gender, name, phone, etc.

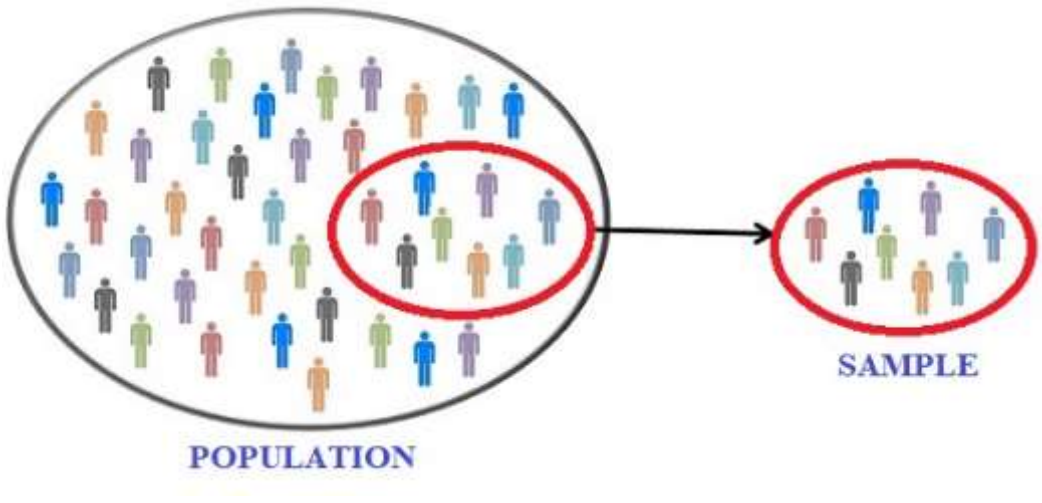
Ordinal Measurement: Examples of ordinal variables include socio-economic status (“low income,” “middle income,” “high income”), education level (“high school,” “BS,” “MS,” “PhD”), income level (“less than 50K,” “50K-100K,” “over 100K”), satisfaction rating (“extremely dislike,” “dislike,” “neutral,” “like,” “extremely like”).

Interval Measurement: An interval scale is one where there is order, and the difference between two values is meaningful. Examples of interval variables include temperature (Fahrenheit and Celsius), pH, SAT score (200-800), and credit score (300-850).

Ratio Measurement: The most common examples of ratio scale are height, money, age, and weight. In market research, observed examples include sales, price, number of customers, and market share.

Population and Sample

A population refers to the complete set of elements or individuals that you aim to make inferences about. Conversely, a sample is a subset of this population from which data is collected. It's important to note that the size of the sample is invariably smaller than the total size of the population. In research, a population isn't confined solely to people; it can encompass various entities such as animal species, geological formations, or even data sets in certain contexts.



Central Tendency

Central tendency (or measure of central tendency) represents a central or typical value within a probability distribution. It serves as a center or point of location within the distribution. Colloquially, measures of central tendency are often referred to as averages. The term 'central tendency' dates back to the late 1920s.

The most commonly used measures of central tendency include the arithmetic mean, the median, and the mode. These measures can be calculated for either a finite set of values or for a theoretical distribution, such as the normal distribution. At times, authors employ 'central tendency' to indicate 'the tendency of quantitative data to cluster around some central value.'

The central tendency of a distribution is typically juxtaposed with its dispersion or variability; dispersion and central tendency often characterize the properties of distributions. Analysis may determine whether data exhibits a strong or weak central tendency based on its dispersion.

In [43]:
```
import numpy as np
import pandas as pd
import seaborn as sns
from scipy import stats

import warnings
warnings.filterwarnings("ignore")
```

In [44]:
```
age  = [23,27,24,23,34,28,23,27,36,38]
```

Mean

In mathematics, particularly in statistics, various types of means exist. The arithmetic mean, often referred to as the expected value or average, represents the central value of a discrete set of numbers. It's calculated by summing all the values in the set and then dividing this sum by the total number of values.

In [45]:
```
mean_age = np.mean(age)
print("Mean:" ,mean_age)
```

```
Mean: 28.3
```

Median

The median serves as the value that divides a dataset, population, or probability distribution into two equal halves: one containing values greater than or equal to the median and the other containing values less than or equal to it. Essentially, it represents 'the middle' value in a dataset.

What distinguishes the median from the mean (often referred to as the 'average') is its resilience against the influence of outliers—extremely large or small values that could skew the data. Because the median is not affected by these outliers, it offers a more robust representation of a 'typical' value within a dataset.

In [46]:
```
median_age = np.median(age)
print("Median:" ,median_age)
```

```
Median: 27.0
```

Mode

The mode represents the value that occurs with the highest frequency in a set of data values. In the context of a discrete random variable X, the mode corresponds to the specific value x (X = x) where the probability mass function achieves its peak, signifying the most probable outcome in the distribution. Essentially, it is the value that is most likely to be sampled or observed within the dataset or probability distribution.

In [47]:
```
mode_age = stats.mode(age)
print("Mode:" ,mode_age[0][0])
```

```
Mode: 23
```

Dispersion

Dispersion, known interchangeably as variability, scatter, or spread, characterizes the degree to which a distribution of data points is stretched or compressed. It quantifies the extent of variability among the values in a dataset or distribution. Measures commonly used to assess statistical dispersion include the variance, standard deviation, and interquartile range.

Range

The range of a set of data is the difference between the largest and smallest values.

In [48]:
```
print("Range: ", (np.max(age)-np.min(age)))
```

```
Range:  15
```

Variance

Variance is the expectation of the squared deviation of a random variable from its mean. Informally, it measures how far a set of numbers is spread out from their average value.

$$\sigma^2 = \frac{\sum_{i=1}^{n}(x_i - \mu)^2}{n}$$

In [49]:
```
print("Variance: ", (np.var(age)))
var = sum((age - np.mean(age))**2)/len(age)
print("Variance with Formula: ", var)
```

```
Variance:  29.21
Variance with Formula:  29.21
```

Standard Deviation

The standard deviation is a measure of the amount of variation or dispersion of a set of values. A low standard deviation indicates that the values tend to be close to the mean (also called the expected value) of the set, while a high standard deviation indicates that the values are spread out over a wider range.

$$\sigma = \sqrt{\frac{\sum(x - \mu)^2}{N}}$$

In [50]:
```
print("Standard Deviation: ", np.std(age))
std = np.sqrt(sum((age - np.mean(age))**2)/len(age))
print("Standard deviation with Formula: ", std)
```

```
Standard Deviation:  5.404627646748664
Standard deviation with Formula:  5.404627646748664
```

Central Tendency and Dispersion

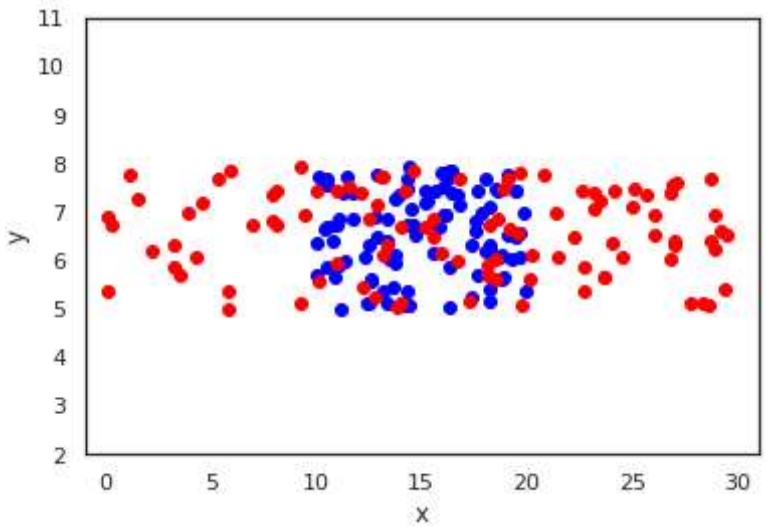
In [51]:
```
import matplotlib.pyplot as plt

y = np.random.uniform(5,8,100)
x1 = np.random.uniform(10,20,100)
x2 = np.random.uniform(0,30,100)

plt.scatter(x1,y,color = "blue")
plt.scatter(x2,y,color = "red")
plt.xlim([-1,31])
plt.ylim([2,11])
plt.xlabel("x")
plt.ylabel("y")
```

```
print("x1 mean: {} and median: {}".format(np.mean(x1),np.median(x1)))
print("x2 mean: {} and median: {}".format(np.mean(x2),np.median(x2)))
```

```
x1 mean: 14.97269139976621 and median: 14.628978436495307
x2 mean: 16.596521802660597 and median: 17.730769029485366
```



In [52]:
```
x1_range = (np.max(x1)-np.min(x1))
x1_variance = (np.var(x1))
x1_std = (np.std(x1))

x2_range = (np.max(x2)-np.min(x2))
x2_variance = (np.var(x2))
x2_std = (np.std(x2))

x1_dispersion = [x1_range, x1_variance, x1_std]
x2_dispersion = [x2_range, x2_variance, x2_std]

df = pd.DataFrame([x1_dispersion,x2_dispersion],columns= ['Range','Variance','Std'], index= ['x1','x2'])
df
```

Out[52]:

	Range	Variance	Std
x1	9.932780	8.793821	2.965438
x2	29.456112	70.966190	8.424143

In [53]:
```
plt.figure(figsize=(6,4))
plt.plot(['Range','Variance','Std'], x1_dispersion, color = "blue")
plt.plot(['Range','Variance','Std'], x2_dispersion, color  = "red")
plt.title("Dispersion", size = 14)
plt.show()
```


Quartiles

Quartile is a type of quantile which divides the number of data points into four parts, or quarters, of more-or-less equal size. The data must be ordered from smallest to largest to compute quartiles; as such, quartiles are a form of order statistic. The three main quartiles are as follows:

- **The first quartile (Q1)** is defined as the middle number between the smallest number (minimum) and the median of the data set. It is also known as the lower or 25th empirical quartile, as **25%** of the data is below this point.
- **The second quartile (Q2)** is the median of a data set; thus **50%** of the data lies below this point.
- **The third quartile (Q3)** is the middle value between the median and the highest value (maximum) of the data set. It is known as the upper or 75th empirical quartile, as **75%** of the data lies below this point.

Along with the minimum and maximum of the data (which are also quartiles), the three quartiles described above provide a five-number summary of the data. This summary is important in statistics because it provides information about both the center and the spread of the data. Knowing the lower and upper quartile provides information on how big the spread is and if the dataset is skewed toward one side. Since quartiles divide the number of data points evenly, the range is not the same between quartiles (i.e., Q3-Q2 ≠ Q2-Q1) and is instead known as the interquartile range (IQR). While the maximum and minimum also show the spread of the data, the upper and lower quartiles can provide more detailed information on the location of specific data points, the presence of outliers in the data, and the difference in spread between the middle 50% of the data and the outer data points.

In [54]:
```
plt.style.use("ggplot")
import os
for dirname, _, filenames in os.walk('/kaggle/input'):
    for filename in filenames:
        print(os.path.join(dirname, filename))
data = pd.read_csv("/kaggle/input/biomechanical-features-of-orthopedic-patients/column_2C_weka.csv")
data.head()
```

```
/kaggle/input/biomechanical-features-of-orthopedic-patients/column_3C_weka.csv
/kaggle/input/biomechanical-features-of-orthopedic-patients/column_2C_weka.csv
/kaggle/input/students-performance-in-exams/StudentsPerformance.csv
```

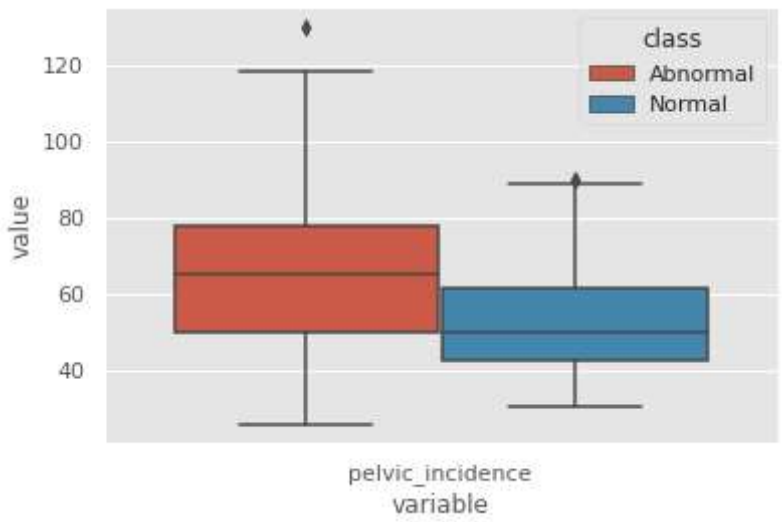
Out[54]:

	pelvic_incidence	pelvic_tilt numeric	lumbar_lordosis_angle	sacral_slope	pelvic_radius	degree_spondylolisthesis	class
0	63.027817	22.552586	39.609117	40.475232	98.672917	-0.254400	Abnormal
1	39.056951	10.060991	25.015378	28.995960	114.405425	4.564259	Abnormal
2	68.832021	22.218482	50.092194	46.613539	105.985135	-3.530317	Abnormal
3	69.297008	24.652878	44.311238	44.644130	101.868495	11.211523	Abnormal
4	49.712859	9.652075	28.317406	40.060784	108.168725	7.918501	Abnormal

In [55]:
```
data_abnormal = data[data["class"] == "Abnormal"]
data_normal = data[data["class"] == "Normal"]
desc = data_abnormal.pelvic_incidence.describe()
Q1 = desc[4]
Q3 = desc[6]
IQR = Q3 - Q1
lower_bound = Q1 - 1.5*IQR
upper_bound = Q3 + 1.5*IQR
print("Anything outside this range is an outlier: (" , lower_bound ,"," , upper_bound,")")
data_abnormal[data_abnormal.pelvic_incidence < lower_bound].pelvic_incidence
print("Outliers: " , data_abnormal[(data_abnormal.pelvic_incidence < lower_bound) | (data_abnormal.pelvic_incidence > u
```

```
Anything outside this range is an outlier: ( 8.865758840000005 , 118.83042036 )
Outliers:  [129.8340406]
```

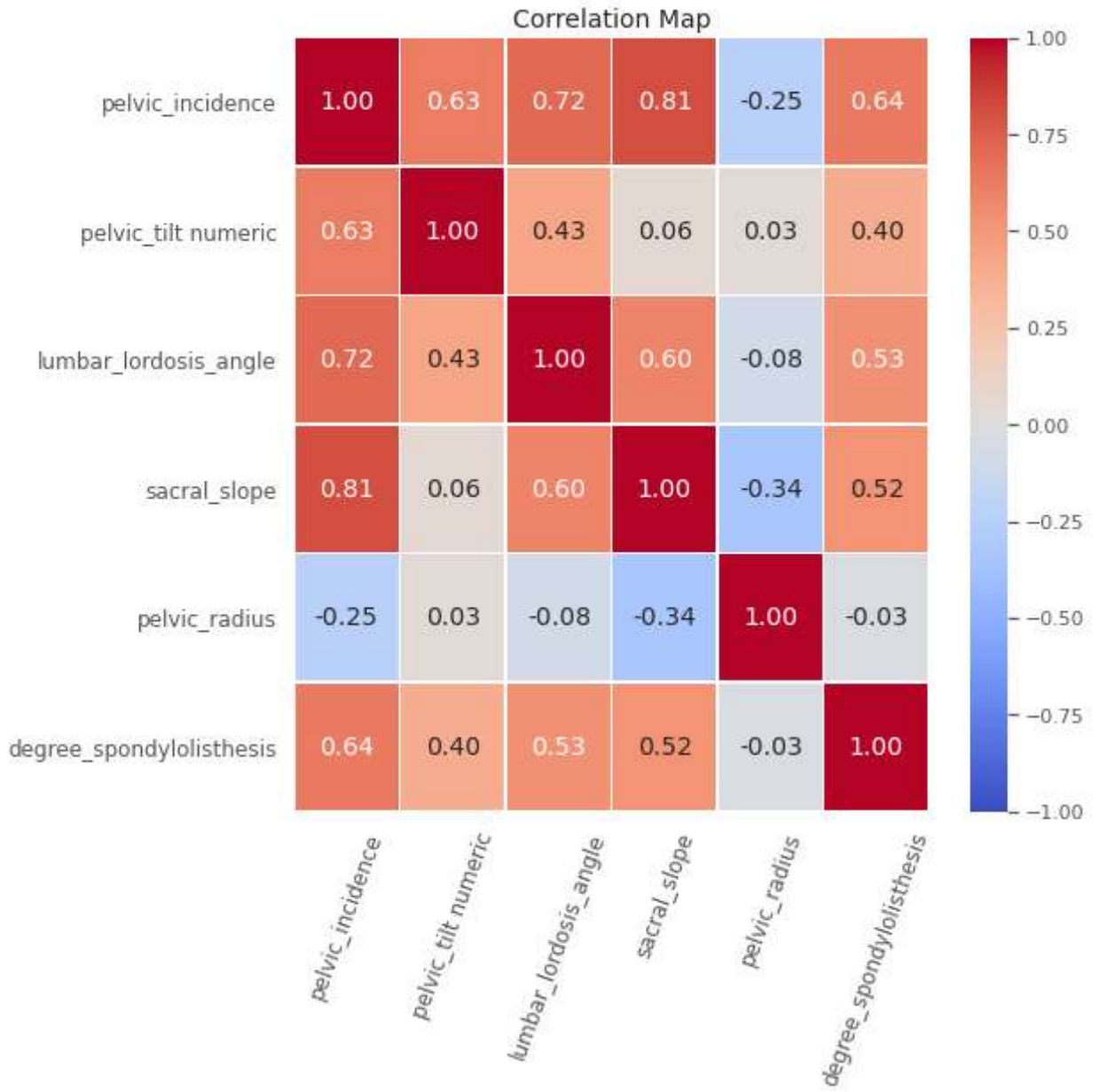
In [56]:
```
melted_data  = pd.melt(data,id_vars = "class", value_vars = ['pelvic_incidence'])
sns.boxplot(x = "variable", y = "value", hue = "class", data = melted_data)
plt.show()
```



Bivariate Data and Covariance

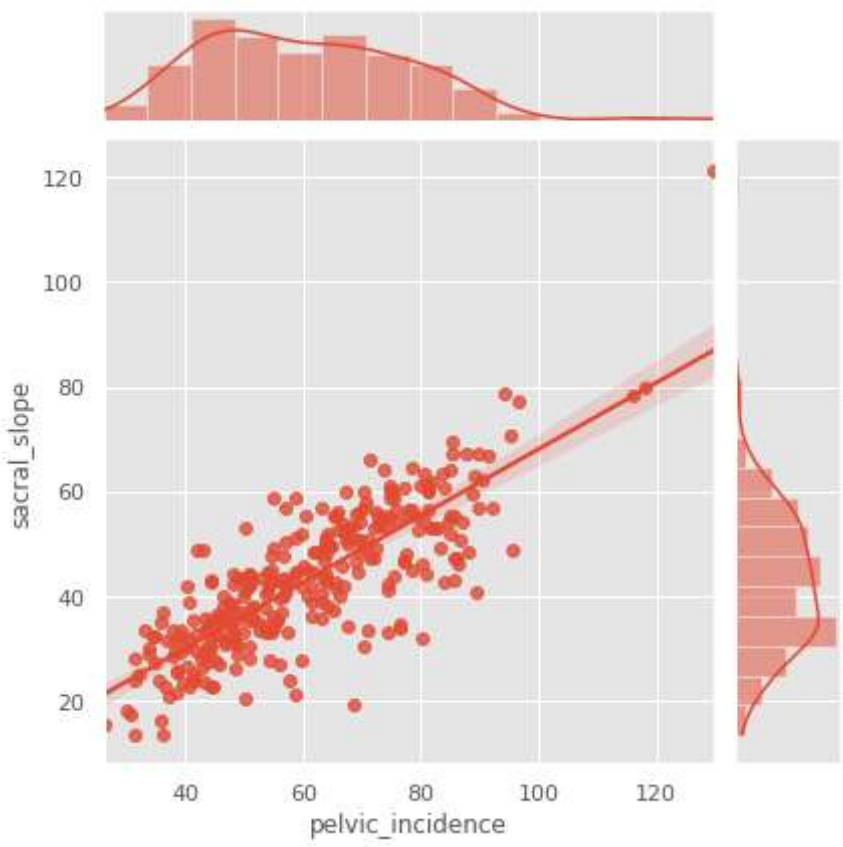
Bivariate data is data on each of two variables, where each value of one of the variables is paired with a value of the other variable. Typically it would be of interest to investigate the possible association between the two variables. The association can be studied via a tabular or graphical display, or via sample statistics which might be used for inference. The method used to investigate the association would depend on the level of measurement of the variable.

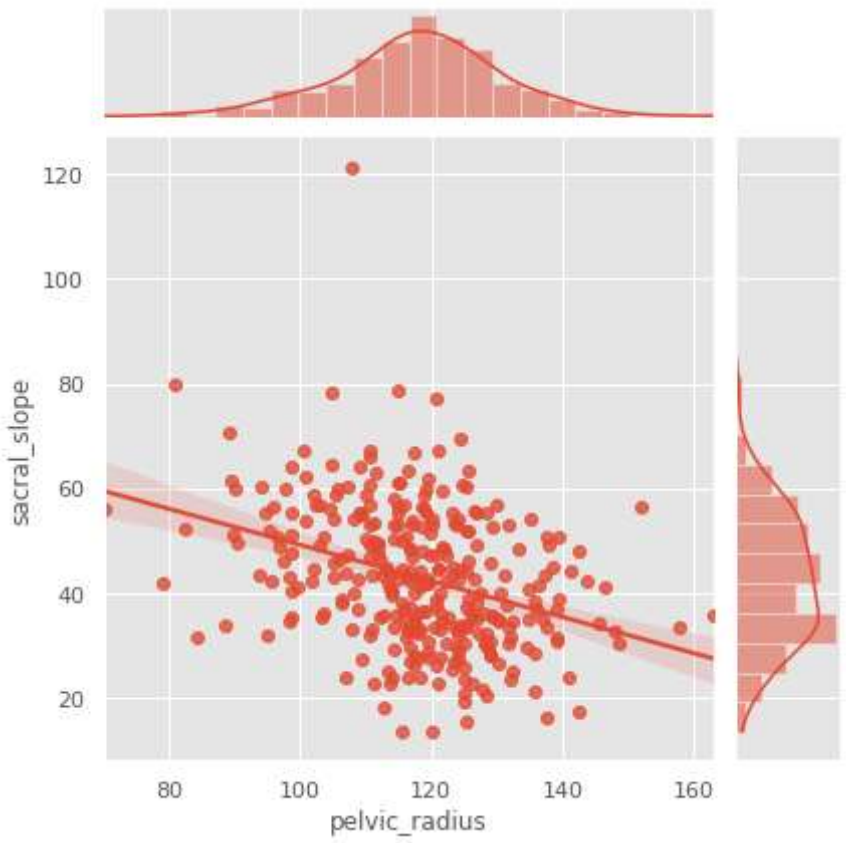
In [57]:
```
f,ax=plt.subplots(figsize = (8,8))
# corr() is actually pearson correlation
sns.heatmap(data.corr(),
            annot= True,
            linewidths=0.5,
            fmt = ".2f",
            vmax = 1,
            vmin = -1,
            ax=ax,
            annot_kws = {'size': 14},
            cmap ="coolwarm")
plt.xticks(rotation=70, size = 12)
plt.yticks(rotation=0, size = 12)
plt.title('Correlation Map',size = 14)
plt.show()
```



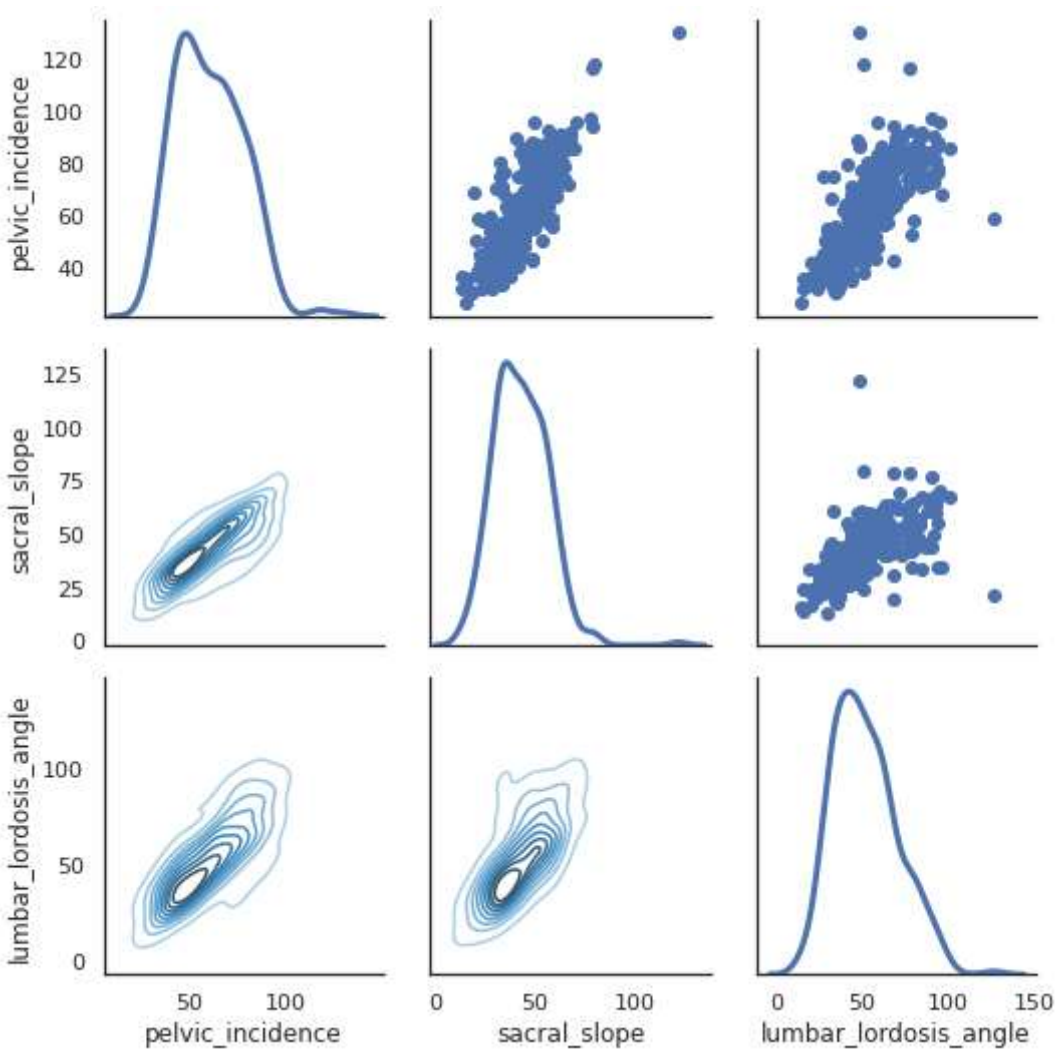
In [58]:
```
plt.figure(figsize = (15,10))
sns.jointplot(data.pelvic_incidence,data.sacral_slope,kind="reg")
sns.jointplot(data.pelvic_radius,data.sacral_slope,kind="reg")
plt.show()
```

```
<Figure size 1080x720 with 0 Axes>
```





In [59]:
```
sns.set(style = "white")
df = data.loc[:,["pelvic_incidence","sacral_slope","lumbar_lordosis_angle"]]
g = sns.PairGrid(df,diag_sharey = False,)
g.map_lower(sns.kdeplot,cmap="Blues_d")
g.map_upper(plt.scatter)
g.map_diag(sns.kdeplot,lw =3)
plt.show()
```

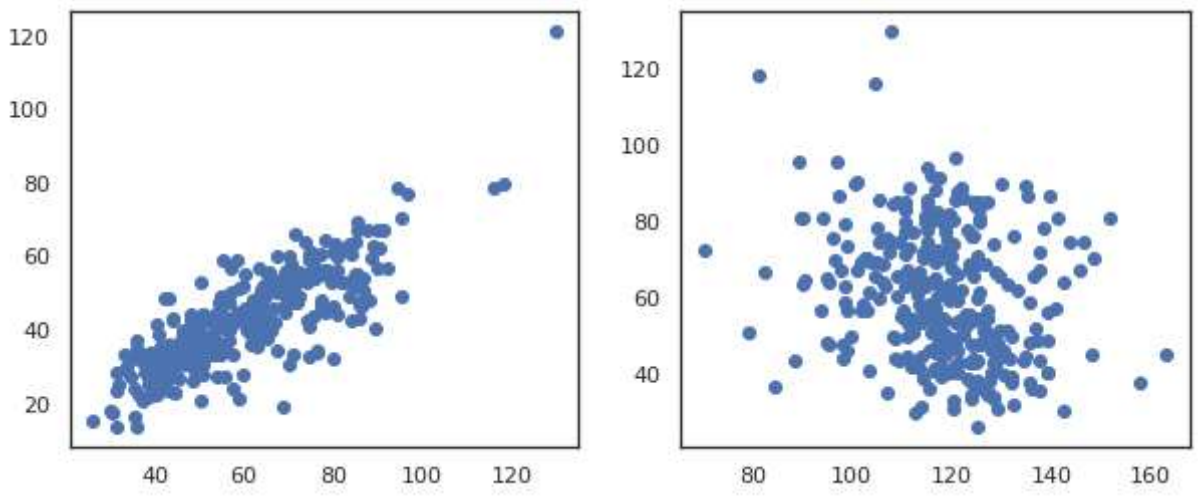


Covariance is a measure of the joint variability of two random variables. If the greater values of one variable mainly correspond with the greater values of the other variable, and the same holds for the lesser values (that is, the variables tend to show similar behavior), the covariance is positive. In the opposite case, when the greater values of one variable mainly correspond to the lesser values of the other, (that is, the variables tend to show opposite behavior), the covariance is negative. The sign of the covariance therefore shows the tendency in the linear relationship between the variables. The magnitude of the covariance is not easy to interpret because it is not normalized and hence depends on the magnitudes of the variables. The normalized version of the covariance, the correlation coefficient, however, shows by its magnitude the strength of the linear relation.

$$cov_{x,y} = \frac{\sum(X_i - \bar{X})(Y_i - \bar{Y})}{n}$$

In [60]:
```
np.cov(data.pelvic_incidence,data.sacral_slope)
print("Covariance between Pelvic Incidence and Sacral Slope: ",data.pelvic_incidence.cov(data.sacral_slope))
print("Covariance between Pelvic Incidence and Pelvic Radius: ",data.pelvic_incidence.cov(data.pelvic_radius))
fig, axs = plt.subplots(1, 2, figsize = (10,4))
axs[0].scatter(data.pelvic_incidence, data.sacral_slope)
axs[1].scatter(data.pelvic_radius, data.pelvic_incidence)
plt.show()
```

```
Covariance between Pelvic Incidence and Sacral Slope:  188.55531498921283
Covariance between Pelvic Incidence and Pelvic Radius:  -56.80491919207533
```



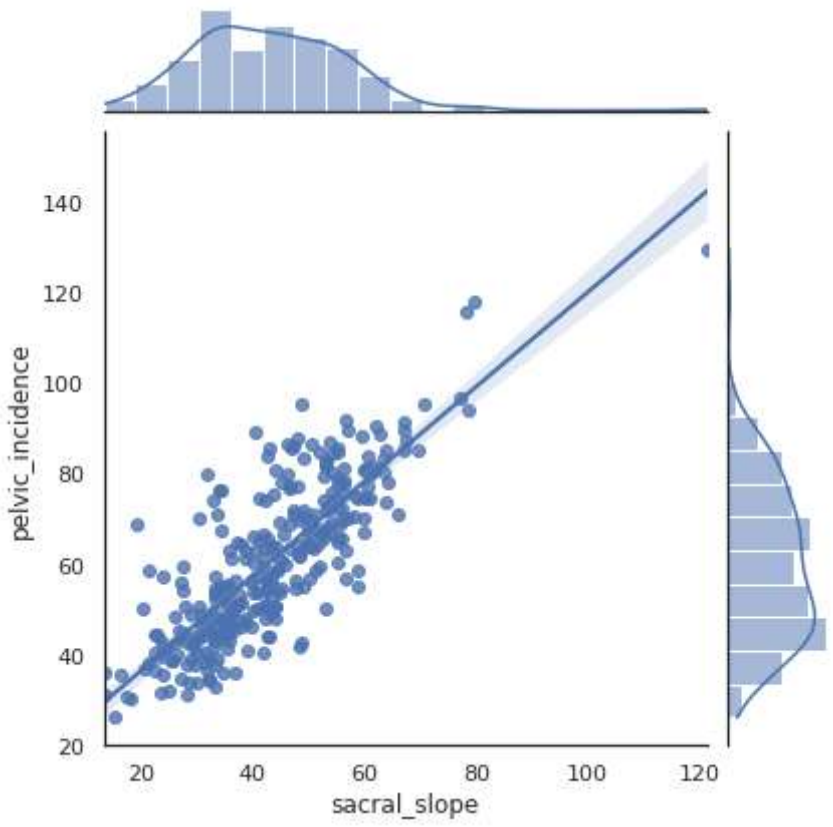
Pearson Correlation Coefficient

Pearson correlation coefficient is the test statistics that measures the statistical relationship, or association, between two continuous variables. It is known as the best method of measuring the association between variables of interest because it is based on the method of covariance. It gives information about the magnitude of the association, or correlation, as well as the direction of the relationship.

In [61]:
```
p1 = data.loc[:,["pelvic_incidence","sacral_slope"]].corr(method= "pearson")
p2 = data.sacral_slope.cov(data.pelvic_incidence)/(data.sacral_slope.std()*data.pelvic_incidence.std())
print('Pearson Correlation: ')
print(p1)
print('Pearson Correlation: ',p2)
```

```
Pearson Correlation:
                  pelvic_incidence  sacral_slope
pelvic_incidence           1.00000       0.81496
sacral_slope               0.81496       1.00000
Pearson Correlation:  0.8149599891850363
```

In [62]:
```
sns.jointplot(data.sacral_slope,data.pelvic_incidence,kind="reg")
plt.show()
```



Spearman Rank Coefficient

The Spearman's rank-order correlation is the nonparametric version of the Pearson product-moment correlation. Spearman's correlation coefficient, (ρ, also signified by rs) measures the strength and direction of association between two ranked variables.

In [63]:
```
ranked_data = data.rank()
spearman_corr = ranked_data.loc[:,["pelvic_incidence","sacral_slope"]].corr(method= "pearson")
print("Spearman's Correlation: ")
print(spearman_corr)
```

```
Spearman's Correlation:
                  pelvic_incidence  sacral_slope
pelvic_incidence           1.00000       0.80083
sacral_slope               0.80083       1.00000
```

Spearman's correlation is little higher than Pearson correlation.

- If relationship between distributions are non linear, spearman's correlation tends to better estimate the strength of relationship.
- Pearson correlation can be affected by outliers. Spearman's correlation is more robust.

Effect Size

Effect size is a number measuring the strength of the relationship between two variables in a statistical population, or a sample-based estimate of that quantity. It can refer to the value of a statistic calculated from a sample of data, the value of a parameter of a hypothetical statistical population, or to the equation that operationalizes how statistics or parameters lead to the effect size value. Examples of effect sizes include the correlation between two variables, the regression coefficient in a regression, the mean difference, or the risk of a particular event (such as a heart attack) happening. Effect sizes complement statistical hypothesis testing, and play an important role in power analyses, sample size planning, and in meta-analyses. The cluster of data-analysis methods concerning effect sizes is referred to as estimation statistics.

$$\frac{M_1 - M_2}{\sqrt{\frac{S_1^2}{n_1} + \frac{S_2^2}{n_2}}}$$

In [64]:
```
mean_diff = data_abnormal.pelvic_incidence.mean() - data_normal.pelvic_incidence.mean()    # m1 - m2
var_abnormal = data_abnormal.pelvic_incidence.var()
var_normal = data_normal.pelvic_incidence.var()
var_pooled = (len(data_abnormal)*var_normal +len(data_normal)*var_abnormal ) / float(len(data_abnormal)+ len(data_norma
effect_size = mean_diff/np.sqrt(var_pooled)
print("Effect Size:",effect_size)
```

```
Effect Size: 0.9101255086409458
```

Probability

Probability is the branch of mathematics concerning numerical descriptions of how likely an event is to occur, or how likely it is that a proposition is true. The probability of an event is a number between 0 and 1, where, roughly speaking, 0 indicates impossibility of the event and 1 indicates certainty. The higher the probability of an event, the more likely it is that the event will occur. A simple example is the tossing of a fair (unbiased) coin. Since the coin is fair, the two outcomes ("heads" and "tails") are both equally probable; the probability of "heads" equals the probability of "tails"; and since no other outcomes are possible, the probability of either "heads" or "tails" is 1/2 (which could also be written as 0.5 or 50%).

Permutation

A permutation of a set is, loosely speaking, an arrangement of its members into a sequence or linear order, or if the set is already ordered, a rearrangement of its elements. The word "permutation" also refers to the act or process of changing the linear order of an ordered set.

Permutations differ from combinations, which are selections of some members of a set regardless of order. For example, written as tuples, there are six permutations of the set {1,2,3}, namely: (1,2,3), (1,3,2), (2,1,3), (2,3,1), (3,1,2), and (3,2,1). These are all the possible orderings of this three-element set. Anagrams of words whose letters are different are also permutations: the letters are already ordered in the original word, and the anagram is a reordering of the letters. The study of permutations of finite sets is an important topic in the fields of combinatorics and group theory.

$$P(n,r) = \frac{n!}{(n-r)!}$$

Permutations are the different ways in which a collection of items can be arranged. For example: The different ways in which the alphabets A, B and C, taken 2 at a time, can be arranged is 3!/(3-2)! = 3!/1! = 6 ways. (AB, AC, BA, BC, CA, CB)

In [65]:
```
import math as math
words = ["A","B","C"]
p = int(math.factorial(len(words)) / math.factorial(len(words)-2))
print(p)
```

```
6
```

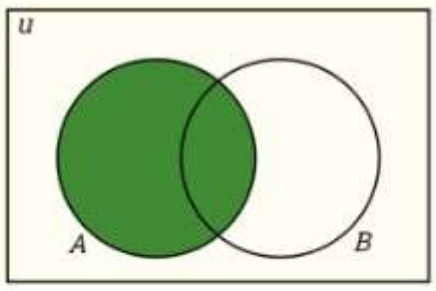
Combination

A combination is a selection of items from a collection, such that the order of selection does not matter (unlike permutations). For example, given three fruits, say an apple, an orange and a pear, there are three combinations of two that can be drawn from this set: an apple and a pear; an apple and an orange; or a pear and an orange. More formally, a k-combination of a set S is a subset of k distinct elements of S. If the set has n elements, the number of k-combinations is equal to the binomial coefficient.

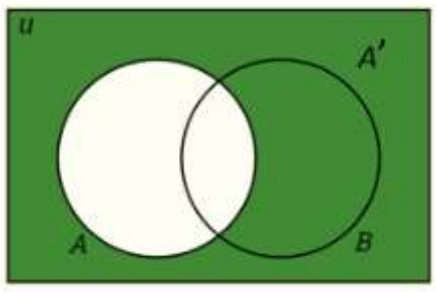
$$C(n, r) = \frac{n!}{r!(n-r)!}$$

Intersection, Unions and Complements

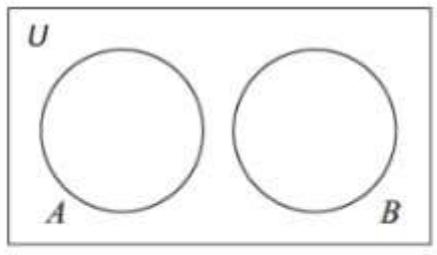
- The **intersection** of two sets A and B, denoted by A ∩ B, is the set containing all elements of A that also belong to B (or equivalently, all elements of B that also belong to A).
- In set theory, the **union** (denoted by ∪) of a collection of sets is the set of all elements in the collection. It is one of the fundamental operations through which sets can be combined and related to each other.



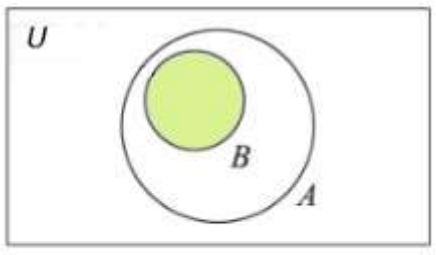
Set *A*



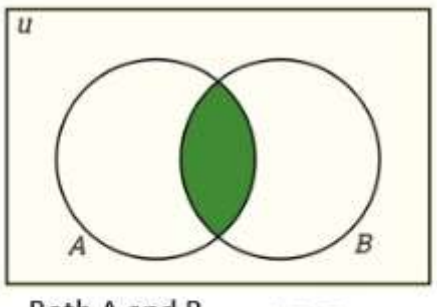
A' the complement of *A*



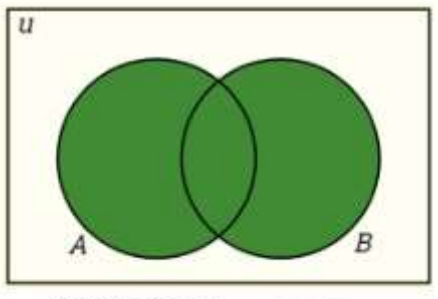
A and *B* are disjoint sets



B is proper subset of *A* $B \subset A$



Both A and B
A intersect *B* $A \cap B$



Either A or B
A union *B* $A \cup B$

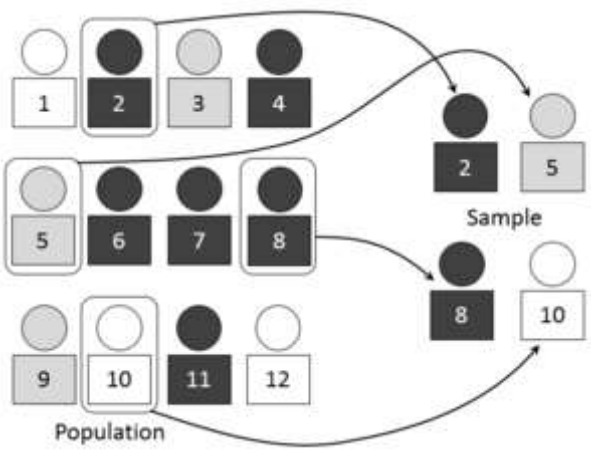
Statistics

Statistics is the discipline that concerns the collection, organization, analysis, interpretation, and presentation of data. In applying statistics to a scientific, industrial, or social problem, it is conventional to begin with a statistical population or a statistical model to be studied. Populations can be diverse groups of people or objects such as "all people living in a country" or "every atom composing a crystal". Statistics deals with every aspect of data, including the planning of data collection in terms of the design of surveys and experiments.

Sampling

Quality assurance, and survey methodology, sampling is the selection of a subset (a statistical sample) of individuals from within a statistical population to estimate characteristics of the whole population. Statisticians attempt for the samples to represent the population in question. Two advantages of sampling are lower cost and faster data collection than measuring the entire population.

Each observation measures one or more properties (such as weight, location, colour) of observable bodies distinguished as independent objects or individuals. In survey sampling, weights can be applied to the data to adjust for the sample design, particularly in stratified sampling. Results from probability theory and statistical theory are employed to guide the practice. In business and medical research, sampling is widely used for gathering information about a population. Acceptance sampling is used to determine if a production lot of material meets the governing specifications.



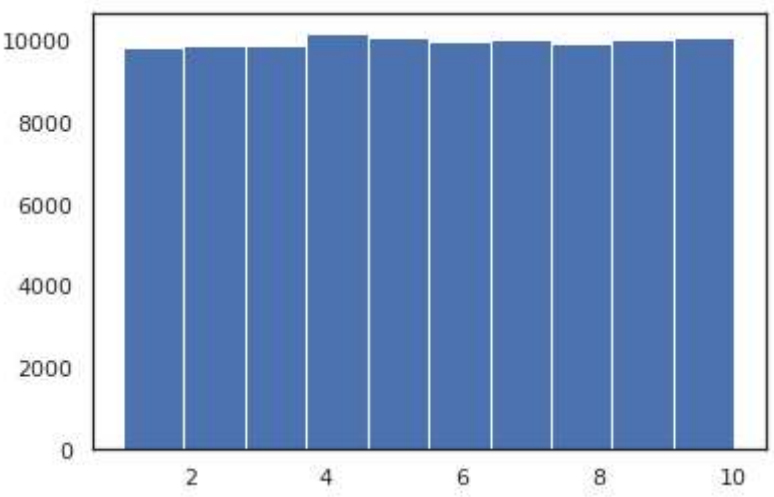
Central Limit Theorem

The central limit theorem states that if you have a population with mean μ and standard deviation σ and take sufficiently large random samples from the population with replacement , then the distribution of the sample means will be approximately normally distributed.

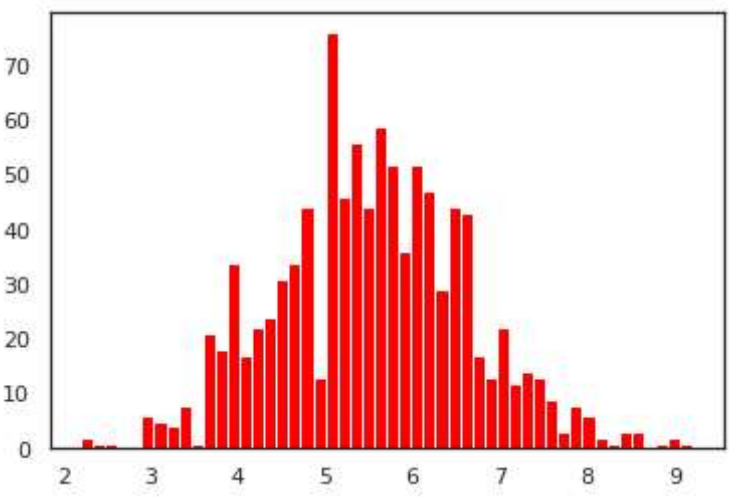
All this is saying is that as you take more samples, especially large ones, your graph of the sample means will look more like a normal distribution.

Here's what the Central Limit Theorem is saying, graphically. The picture below shows one of the simplest types of test: rolling a fair die. The more times you roll the die, the more likely the shape of the distribution of the means tends to look like a normal distribution graph.

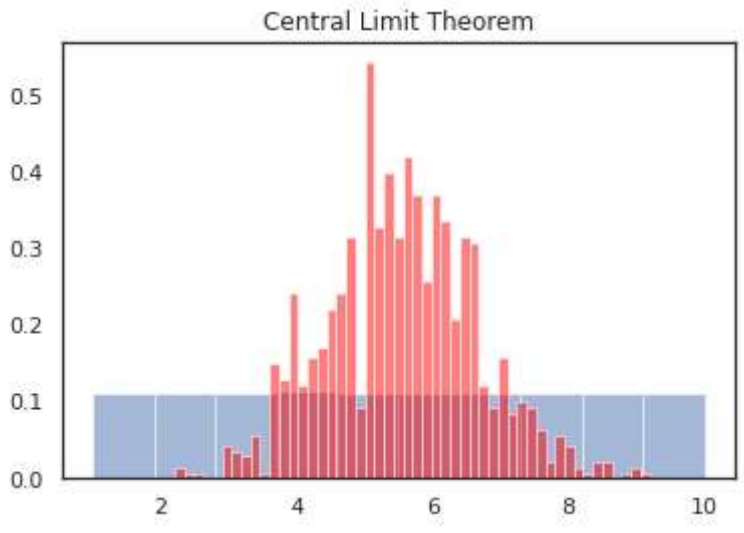
In [66]:
```
x = np.random.random_integers(10,size=100000)
plt.hist(x)
plt.show()
```



In [67]:
```
import random
mean_sample = []
for i in range(1000):
    sample = random.randrange(5,10)
    mean_sample.append(np.mean(random.sample(list(x),sample)))
plt.hist(mean_sample,bins = 50, color = "red")
plt.show()
```



In [68]:
```
plt.hist(x,alpha = 0.5,density=True)
plt.hist(mean_sample,bins = 50,alpha = 0.5,color = "red",density=True)
plt.title("Central Limit Theorem")
plt.show()
```



Standard Error

The standard error (SE) of a statistic (usually an estimate of a parameter) is the standard deviation of its sampling distribution or an estimate of that standard deviation. If the statistic is the sample mean, it is called the standard error of the mean (SEM).

The sampling distribution of a mean is generated by repeated sampling from the same population and recording of the sample means obtained. This forms a distribution of different means, and this distribution has its own mean and variance. Mathematically, the variance of the sampling distribution obtained is equal to the variance of the population divided by the sample size. This is because as the sample size increases, sample means cluster more closely around the population mean.

Therefore, the relationship between the standard error of the mean and the standard deviation is such that, for a given sample size, the standard error of the mean equals the standard deviation divided by the square root of the sample size. In other words, the standard error of the mean is a measure of the dispersion of sample means around the population mean.

In regression analysis, the term "standard error" refers either to the square root of the reduced chi-squared statistic, or the standard error for a particular regression coefficient (as used in, say, confidence intervals).

Hypothesis Testing

Hypothesis testing refers to the process of making inferences or educated guesses about a particular parameter. This can either be done using statistics and sample data, or it can be done on the basis of an uncontrolled observational study.

When a predetermined number of subjects in a hypothesis test prove the "alternative hypothesis," then the original hypothesis (the "null hypothesis") is overturned or "rejected." You must decide the level of statistical significance in your hypothesis, as you can never be 100 percent confident in your findings. First, let's examine the steps to test a hypothesis. Then, we'll enjoy some examples of hypothesis testing.

How to test a Hypothesis?

At this point, you'll already have a hypothesis ready to go. Now, it's time to test your theory. Remember, a hypothesis is a statement regarding what you believe might happen. These are the steps you'll want to take to see if your suppositions stand up:

1. **State your null hypothesis.** The null hypothesis is a commonly accepted fact. It's the default, or what we'd believe if the experiment was never conducted. It's the least exciting result, showing no significant difference between two or more groups. Researchers work to nullify or disprove null hypotheses.
2. **State an alternative hypothesis.** You'll want to prove an alternative hypothesis. This is the opposite of the null hypothesis, demonstrating or supporting a statistically significant result. By rejecting the null hypothesis, you accept the alternative hypothesis.
3. **Determine a significance level.** This is the determiner, also known as the alpha (α). It defines the probability that the null hypothesis will be rejected. A typical significance level is set at 0.05 (or 5%). You may also see 0.1 or 0.01, depending on the area of study. If you set the alpha at 0.05, then there is a 5% chance you'll find support for the alternative hypothesis (thus rejecting the null hypothesis) when, in truth, the null hypothesis is actually true and you were wrong to reject it. In other words, the significance level is a statistical way of demonstrating how confident you are in your conclusion. If you set a high alpha (0.25), then you'll have a better shot at supporting your alternative hypothesis, since you don't need to find as big a difference between your test groups. However, you'll also have a bigger chance at being wrong about your conclusion.
4. **Calculate the p-value.** The p-value, or calculated probability, indicates the probability of achieving the results of the null hypothesis. While the alpha is the significance level you're trying to achieve, the p-level is what your actual data is showing when you calculate it. A low p-value offers stronger support for your alternative hypothesis.
5. **Draw a conclusion.** If your p-value meets your significance level requirements, then your alternative hypothesis may be valid and you may reject the null hypothesis. In other words, if your p-value is less than your significance level (e.g., if your calculated p-value is 0.02 and your significance level is 0.05), then you can reject the null hypothesis and accept your alternative hypothesis.

Hypothesis Testing Example

Let's take those five steps and look at a couple of real-world scenarios.

Peppermint Essential Oil

Essential oils are becoming more and more popular. Chamomile, lavender, and ylang-ylang are commonly touted as anxiety remedies. Perhaps you'd like to test the healing powers of peppermint essential oil. Your hypothesis might go something like this:

1. **Null hypothesis** - Peppermint essential oil has no effect on the pangs of anxiety.
2. **Alternative hypothesis** - Peppermint essential oil alleviates the pangs of anxiety.
3. **Significance level** - The significance level is 0.25 (allowing for a better shot at proving your alternative hypothesis).
4. **P-value** - The p-value is calculated as 0.05.
5. **Conclusion** - After providing one group with peppermint oil and the other with a placebo, you gauge the difference between the two based on self-reported levels of anxiety. Based on your calculations, the difference between the two groups is statistically significant with a p-value of 0.05, well below the defined alpha of 0.25. You conclude that your study supports the alternative hypothesis that peppermint essential oil can alleviate the pangs of anxiety.

Hypothesis Testing with Our Data

Null hypothesis: relationship between pelvic incidence and sacral slope is zero.
Alternate hypothesis: relationship between pelvic incidence and sacral slope is not zero.

Let's find p-value.

In [69]:
```
statistic, p_value = stats.ttest_rel(data.pelvic_incidence,data.sacral_slope)
p_value = round(p_value,3)
print('p-value: ',p_value)
if p_value == 0:
    print("Reject null hypothesis, alternate hypothesis is correct, relationship between pelvic incidence and sacral slo
else:
    print("Fail to reject null hypothesis, relationship between pelvic incidence and sacral slope is zero.")
```

```
p-value:  0.0
Reject null hypothesis, alternate hypothesis is correct, relationship between pelvic incidence and sacral slope is not
zero.
```

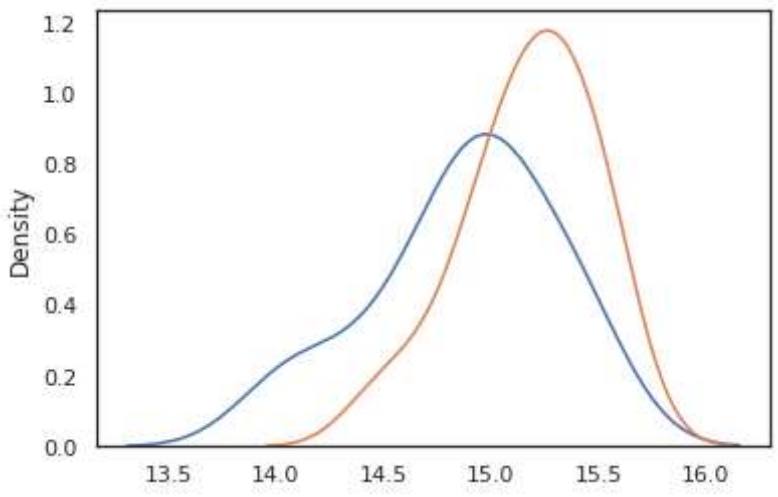
T-Distribution

Student's t-distribution (or simply the t-distribution) is any member of a family of continuous probability distributions that arise when estimating the mean of a normally-distributed population in situations where the sample size is small and the population's standard deviation is unknown. It was developed by English statistician William Sealy Gosset under the pseudonym "Student".

The t-distribution plays a role in a number of widely used statistical analyses, including Student's t-test for assessing the statistical significance of the difference between two sample means, the construction of confidence intervals for the difference between two population means, and in linear regression analysis. The Student's t-distribution also arises in the Bayesian analysis of data from a normal family.

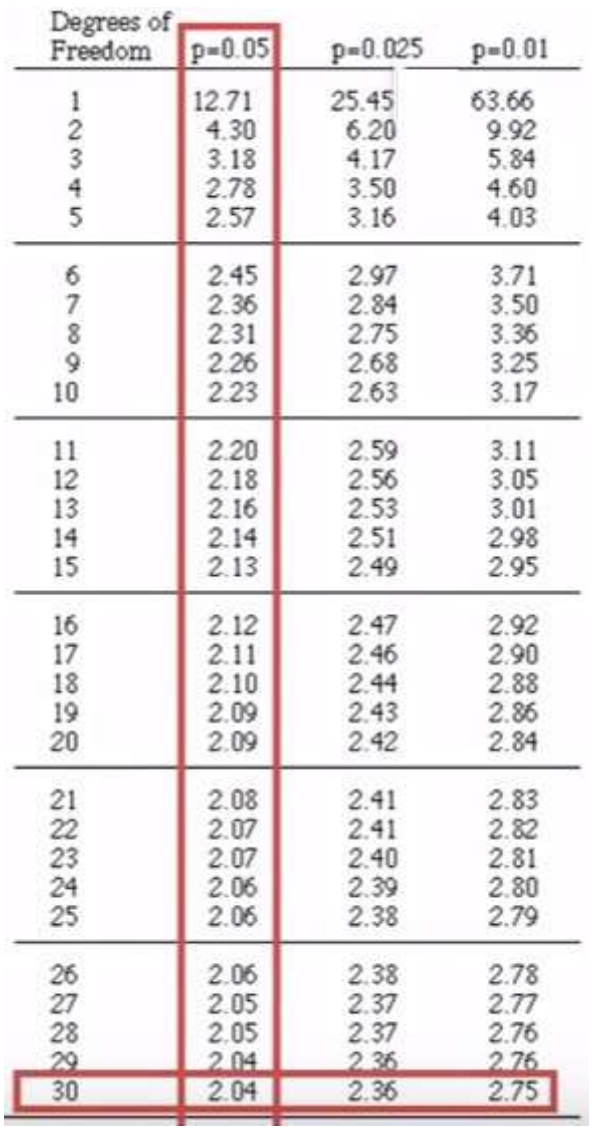
In [70]:
```
s1 = np.array([14.67230258, 14.5984991 , 14.99997003, 14.83541808, 15.42533116,
       15.42023888, 15.0614731 , 14.43906856, 15.40888636, 14.87811941,
       14.93932134, 15.04271942, 14.96311939, 14.0379782 , 14.10980817,
       15.23184029])
print("mean 1: ", np.mean(s1))
print("standart deviation 1: ", np.std(s1))
print("variance 1: ", np.var(s1))
s2 = np.array([15.23658167, 15.30058977, 15.49836851, 15.03712277, 14.72393502,
       14.97462198, 15.0381114 , 15.18667258, 15.5914418 , 15.44854406,
       15.54645152, 14.89288726, 15.36069141, 15.18758271, 14.48270754,
       15.28841374])
print("mean 2: ", np.mean(s2))
print("standart deviation 2: ", np.std(s2))
print("variance 2: ", np.var(s2))
# visualize with pdf
import seaborn as sns
sns.kdeplot(s1)
sns.kdeplot(s2)
plt.show()
```

```
mean 1:  14.879005879375
standart deviation 1:  0.4106367159180011
variance 1:  0.16862251245992116
mean 2:  15.17467023375
standart deviation 2:  0.29561498265957253
variance 2:  0.08738821797281937
```



In [71]:
```
t_val = np.abs(np.mean(s1)-np.mean(s2))/np.sqrt((np.var(s1)/len(s1))+(np.var(s2)/len(s2)))
print("t-value: ", t_val)
```

```
t-value:  2.3373829708002227
```



Degrees of Freedom	p=0.05	p=0.025	p=0.01
1	12.71	25.45	63.66
2	4.30	6.20	9.92
3	3.18	4.17	5.84
4	2.78	3.50	4.60
5	2.57	3.16	4.03
6	2.45	2.97	3.71
7	2.36	2.84	3.50
8	2.31	2.75	3.36
9	2.26	2.68	3.25
10	2.23	2.63	3.17
11	2.20	2.59	3.11
12	2.18	2.56	3.05
13	2.16	2.53	3.01
14	2.14	2.51	2.98
15	2.13	2.49	2.95
16	2.12	2.47	2.92
17	2.11	2.46	2.90
18	2.10	2.44	2.88
19	2.09	2.43	2.86
20	2.09	2.42	2.84
21	2.08	2.41	2.83
22	2.07	2.41	2.82
23	2.07	2.40	2.81
24	2.06	2.39	2.80
25	2.06	2.38	2.79
26	2.06	2.38	2.78
27	2.05	2.37	2.77
28	2.05	2.37	2.76
29	2.04	2.36	2.76
30	2.04	2.36	2.75

In [72]:
```
critical_value = 2.04
print("Null hypothesis: There is no statistically significant difference between these two distributions.")
if t_val > critical_value:
    print("t value > critical value")
    print("Reject Null Hypothesis")
else:
    print("t value < critical value")
    print("Fail to reject Null Hypothesis ")
```

```
Null hypothesis: There is no statistically significant difference between these two distributions.
t value > critical value
Reject Null Hypothesis
```

A/B Testing

A/B testing (also known as bucket testing or split-run testing) is a user experience research methodology. A/B tests consist of a randomized experiment with two variants, A and B. It includes application of statistical hypothesis testing or "two-sample hypothesis testing" as used in the field of statistics. A/B testing is a way to compare two versions of a single variable, typically by testing a subject's response to variant A against variant B, and determining which of the two variants is more effective.

In [73]:
```
from scipy.stats import shapiro,levene
data = pd.read_csv("../input/students-performance-in-exams/StudentsPerformance.csv")
data[data['parental level of education'].isin(["bachelor's degree",'high school'])]. \
groupby('parental level of education').agg({'math score':'mean'})
```

Out[73]:

parental level of education	math score
bachelor's degree	69.389831
high school	62.137755

Shapiro-Wilks Normality Test

The normality assumption is more important when the two groups have small sample sizes than for larger sample sizes.

Normal distributions are symmetric, which means they are “even” on both sides of the center. Normal distributions do not have extreme values, or outliers. You can check these two features of a normal distribution with graphs. Earlier, we decided that the body fat data was “close enough” to normal to go ahead with the assumption of normality. The figure below shows a normal quantile plot for men and women, and supports our decision.

In [74]:
```
test_stat, p = shapiro(data[data['parental level of education'] == "bachelor's degree"]['math score'])
print('Test Stat: {}'.format(round(test_stat,4)))
print('p-value: {}'.format(round(p,4)))
if p < 0.05:
    print('p < 0.05 --> Reject Null Hypothesis, data are not normally distributed.')
else:
    print('p > 0.05 --> Cannot reject Null Hypothesis, data are normally distributed.')
```

```
Test Stat: 0.9906
p-value: 0.6043
p > 0.05 --> Cannot reject Null Hypothesis, data are normally distributed.
```

In [75]:
```
test_stat, p = shapiro(data[data['parental level of education'] == 'high school']['math score'])
print('Test Stat: {}'.format(round(test_stat,4)))
print('p-value: {}'.format(round(p,4)))
if p < 0.05:
    print('p < 0.05 --> Reject Null Hypothesis, data are not normally distributed.')
else:
    print('p > 0.05 --> Cannot reject Null Hypothesis, data are normally distributed.')
```

```
Test Stat: 0.9868
p-value: 0.0652
p > 0.05 --> Cannot reject Null Hypothesis, data are normally distributed.
```

Levene Test for Equality of Variances

Levene's test is used to test if k samples have equal variances. Equal variances across samples is called homogeneity of variance. Some statistical tests, for example the analysis of variance, assume that variances are equal across groups or samples. The Levene test can be used to verify that assumption.

In [76]:
```
test_stat, p = levene(data[data['parental level of education'] == "bachelor's degree"]['math score'],
                      data[data['parental level of education'] == 'high school']['math score'])

print('Test Stat: {}'.format(round(test_stat,4)))
print('p-value: {}'.format(round(p,4)))

if p < 0.05:
    print('p < 0.05 --> Reject Null Hypothesis, variances are not equal.')
else:
    print('p > 0.05 --> Cannot reject Null Hypothesis, variances are equal.')
```

```
Test Stat: 0.2283
p-value: 0.6331
p > 0.05 --> Cannot reject Null Hypothesis, variances are equal.
```

Two-Samples T-Test

The two-sample t-test (also known as the independent samples t-test) is a method used to test whether the unknown population means of two groups are equal or not.

Is this the same as an A/B test?
Yes, a two-sample t-test is used to analyze the results from A/B tests.

When can I use the test?
You can use the test when your data values are independent, are randomly sampled from two normal populations and the two independent groups have equal variances.

In [77]:
```
test_stat, p = stats.ttest_ind(data[data['parental level of education'] == "bachelor's degree"]['math score'],
                               data[data['parental level of education'] == 'high school']['math score'],
                               equal_var = True)

print('Test Stat: {}'.format(round(test_stat,4)))
print('p-value: {}'.format(round(p,4)))

if p < 0.05:
    print('p < 0.05 --> Reject Null Hypothesis, population means are not the same.')

else:
    print('p > 0.05 --> Cannot reject Null Hypothesis, population means are the same.')
```

```
Test Stat: 4.2361
p-value: 0.0
p < 0.05 --> Reject Null Hypothesis, population means are not the same.
```

ANOVA (Analysis of Variance)

Analysis of variance (ANOVA) is an analysis tool used in statistics that splits an observed aggregate variability found inside a data set into two parts: systematic factors and random factors. The systematic factors have a statistical influence on the given data set, while the random factors do not. Analysts use the ANOVA test to determine the influence that independent variables have on the dependent variable in a regression study.

The t- and z-test methods developed in the 20th century were used for statistical analysis until 1918, when Ronald Fisher created the analysis of variance method. ANOVA is also called the Fisher analysis of variance, and it is the extension of the t- and z-tests. The term became well-known in 1925, after appearing in Fisher's book, "Statistical Methods for Research Workers." It was employed in experimental psychology and later expanded to subjects that were more complex.

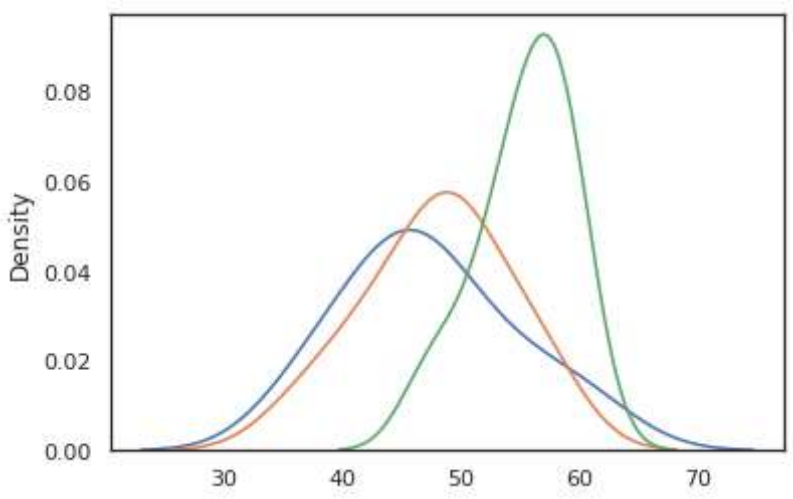
- For example, are the exam anxiety of middle school, high school and university students different from each other? We will answer the question with ANOVA.
- Null Hypothesis: Exam concerns same

In [78]:
```
middle_school = np.array([51.36372405, 44.96944041, 49.43648441, 45.84584407, 45.76670682,
       56.04033356, 60.85163656, 39.16790361, 36.90132329, 43.58084076])
high_school = np.array([56.65674765, 55.92724431, 42.32435143, 50.19137162, 48.91784081,
       48.11598035, 50.91298812, 47.46134988, 42.76947742, 36.86738678])
university = np.array([60.03609029, 56.94733648, 57.77026852, 47.29851926, 54.21559389,
       57.74008243, 50.92416154, 53.47770749, 55.62968872, 59.42984391])

print("Middle school Mean: ",np.mean(middle_school))
print("High school Mean: ",np.mean(high_school))
print("University Mean: ",np.mean(university))
total_mean = (np.mean(middle_school) + np.mean(high_school) + np.mean(university))/3
print("Total Mean: ",np.mean(total_mean))


sns.kdeplot(middle_school)
sns.kdeplot(high_school)
sns.kdeplot(university)
plt.show()
```

```
Middle school Mean:  47.392423754
High school Mean:  48.014473837
University Mean:  55.346929253000006
Total Mean:  50.25127561466667
```



In [79]:
```
stats.f_oneway(middle_school, high_school, university)
```

Out[79]:
```
F_onewayResult(statistic=5.5230098837341215, pvalue=0.009755200989550988)
```

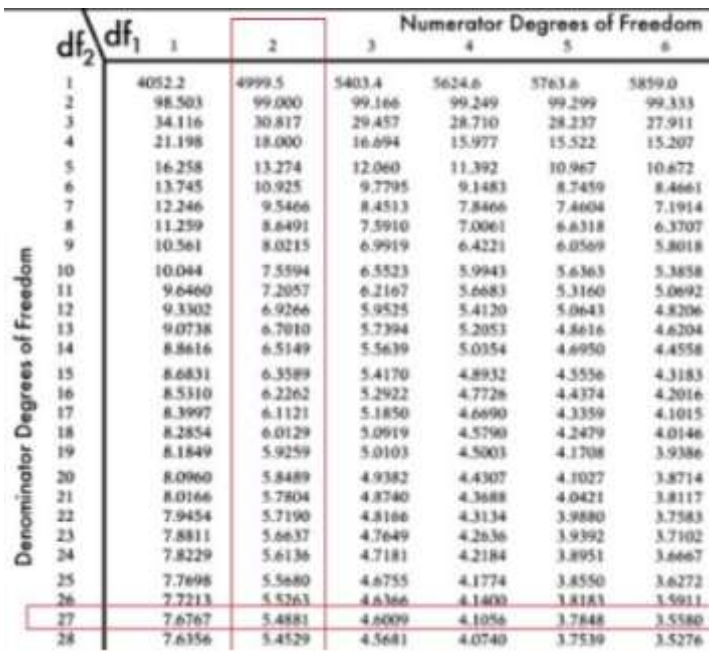
In [80]:
```
f_value = stats.f_oneway(middle_school, high_school, university)[0]
print("F value:", f_value)
```

```
F value: 5.5230098837341215
```

F-Distribution

In probability theory and statistics, the F-distribution, also known as Snedecor's F distribution or the Fisher–Snedecor distribution (after Ronald Fisher and George W. Snedecor) is a continuous probability distribution that arises frequently as the null distribution of a test statistic, most notably in the analysis of variance (ANOVA), e.g., F-test.

- F value < critical value --> fail to reject null hypothesis
- F value > critical value --> reject null hypothesis
- Degrees of freedom for groups: Number of groups - 1
  3 - 1 = 2
- Degrees of freedom for error: (number of rows - 1) *number of groups*
  (10 - 1) 3 = 27



df_2 \ df_1	Numerator Degrees of Freedom 1	2	3	4	5	6
1	4052.2	4999.5	5403.4	5624.6	5763.6	5859.0
2	98.503	99.000	99.166	99.249	99.299	99.333
3	34.116	30.817	29.457	28.710	28.237	27.911
4	21.198	18.000	16.694	15.977	15.522	15.207
5	16.258	13.274	12.060	11.392	10.967	10.672
6	13.745	10.925	9.7795	9.1483	8.7459	8.4661
7	12.246	9.5466	8.4513	7.8466	7.4604	7.1914
8	11.259	8.6491	7.5910	7.0061	6.6318	6.3707
9	10.561	8.0215	6.9919	6.4221	6.0569	5.8018
10	10.044	7.5594	6.5523	5.9943	5.6363	5.3858
11	9.6460	7.2057	6.2167	5.6683	5.3160	5.0692
12	9.3302	6.9266	5.9525	5.4120	5.0643	4.8206
13	9.0738	6.7010	5.7394	5.2053	4.8616	4.6204
14	8.8616	6.5149	5.5639	5.0354	4.6950	4.4558
15	8.6831	6.3589	5.4170	4.8932	4.5556	4.3183
16	8.5310	6.2262	5.2922	4.7726	4.4374	4.2016
17	8.3997	6.1121	5.1850	4.6690	4.3359	4.1015
18	8.2854	6.0129	5.0919	4.5790	4.2479	4.0146
19	8.1849	5.9259	5.0103	4.5003	4.1708	3.9386
20	8.0960	5.8489	4.9382	4.4307	4.1027	3.8714
21	8.0166	5.7804	4.8740	4.3688	4.0421	3.8117
22	7.9454	5.7190	4.8166	4.3134	3.9880	3.7583
23	7.8811	5.6637	4.7649	4.2636	3.9392	3.7102
24	7.8229	5.6136	4.7181	4.2184	3.8951	3.6667
25	7.7698	5.5680	4.6755	4.1774	3.8550	3.6272
26	7.7213	5.5263	4.6366	4.1400	3.8183	3.5911
27	7.6767	5.4881	4.6009	4.1056	3.7848	3.5580
28	7.6356	5.4529	4.5681	4.0740	3.7539	3.5276

Denominator Degrees of Freedom

In [81]:
```
critical_value = 5.4881
if f_value > critical_value:
    print("Reject to Null Hypothesis (f_value > critical_value)")
else:
    print("Fail to reject Null Hypothesis (critical_value > f_value)")
```

```
Reject to Null Hypothesis (f_value > critical_value)
```

Chi-Square Analysis

A chi-squared test, also written as χ2 test, is a statistical hypothesis test that is valid to perform when the test statistic is chi-squared distributed under the null hypothesis, specifically Pearson's chi-squared test and variants thereof. Pearson's chi-squared test is used to determine whether there is a statistically significant difference between the expected frequencies and the observed frequencies in one or more categories of a contingency table.

In the standard applications of this test, the observations are classified into mutually exclusive classes. If the null hypothesis that there are no differences between the classes in the population is true, the test statistic computed from the observations follows a χ2 frequency distribution. The purpose of the test is to evaluate how likely the observed frequencies would be assuming the null hypothesis is true.

Test statistics that follow a χ2 distribution occur when the observations are independent and normally distributed, which assumptions are often justified under the central limit theorem. There are also χ2 tests for testing the null hypothesis of independence of a pair of random variables based on observations of the pairs.

Chi-squared tests often refers to tests for which the distribution of the test statistic approaches the χ2 distribution asymptotically, meaning that the sampling distribution (if the null hypothesis is true) of the test statistic approximates a chi-squared distribution more and more closely as sample sizes increase.

For example, let's give an example, we throw money in the air 10 times. It comes to 9 tails and 1 head.

- Our question is: 9 times there is no chance of tails or if this money is inclined to tails? so is it biased (you can also consider it fraudulent)
- Null hypothesis: For a fair coin, it makes sense to get 9 tails out of 10 shots with a statistically 95% probability (confidence level 0.05).

For the tails in our example:

- expected frequency = 5
- observed frequency = 9

For heads:

- expected frequency = 5
- observed frequency = 1

Chi-Sqaure Value: 6.4

	Probability of exceeding the critical value						
d	0.05	0.01	0.001	*d*	0.05	0.01	0.001
1	3.841	6.635	10.828	11	19.675	24.725	31.264
2	5.991	9.210	13.816	12	21.026	26.217	32.910
3	7.815	11.345	16.266	13	22.362	27.688	34.528

If the chi-square value is less than the critical value, there is a high correlation between observation and expected values.

- 6.4> 3.8 so reject the null hypothesis.

Chi-Square Analysis Example

There are 7 washing machines with equal probability of deterioration in the laundry. So expected = failure rate should be same for all washing machines.

- Washing machines are independent of each other.
- Observations: 1(5), 2(7), 3(9), 4(4), 5(1), 6(10), 7(6)
- Null Hypothesis: observation values in this way makes sense with a statistically 95% probability.
- Total Deterioration: 42
- Expected Value: 42 / 7 = 6
- Degrees of Freedom: 7 - 1 = 6

In [82]:
```
observation = np.array([5,7,9,4,1,10,6])
print("Total: ",np.sum(observation))
expected = np.sum(observation)/ len(observation)
print("Expected: ",expected)
chi_value = np.sum(((observation - expected)**2)/expected)
print("Chi_value: ",chi_value)
```

```
Total:  42
Expected:  6.0
Chi_value:  9.333333333333334
```

In [83]:
```
from scipy.stats import chi2
crit_value = chi2.isf(0.05,6)
print("Critical value: ", crit_value)
```

```
Critical value:  12.59158724374398
```

In [84]:
```
if crit_value > chi_value:
    print("Fail to reject Null Hypothesis (crit_value > chi_value)")
else:
    print("Reject Null Hypothesis (chi_value > crit_value)")
```

```
Fail to reject Null Hypothesis (crit_value > chi_value)
```