

# Applied Statistical Methods

## Tests of Means of Numerical Data- Part II (fixed effects ANOVA)

Xuemao Zhang  
East Stroudsburg University

February 24, 2023

# Outline

- Design of Experiments
- Analysis of Variance
  - ▶ One-Way ANOVA
  - ▶ Two-Way ANOVA
    - ★ Complete Block Design
    - ★ Factorial Design
  - ▶ Three-Way ANOVA

# Design of Experiments

- Experiment: people apply some treatment and then observe its effects on the subjects (subjects in an experiments generally are called experimental units)
  - ▶ An experiment requires random assignment of subjects to treatments.
  - ▶ If done correctly, experiments provide most compelling evidence that a treatment causes an observed outcome
  - ▶ For example, in a randomized clinical study patients in the experimental groups receive the drug while patients in the control groups receive a placebo or sugar pill. The patients do not know if they are receiving the experimental treatment or placebo.

# Design of Experiments

- Some terminologies:
  - ▶ An experimental unit is the object on which a measurement (or measurements) is taken.
  - ▶ The response is the variable being measured by the experimenter.
  - ▶ A factor is an independent variable whose values are controlled and varied by the experimenter.
  - ▶ A level is the intensity setting of a factor.
  - ▶ A treatment is a specific combination of factor levels.

# Design of Experiments

- Example: Is the attention span of children affected by whether or not they had a good breakfast? Twelve children were randomly divided into three groups and assigned to a different meal plan. The response was attention span in minutes during the morning reading time.

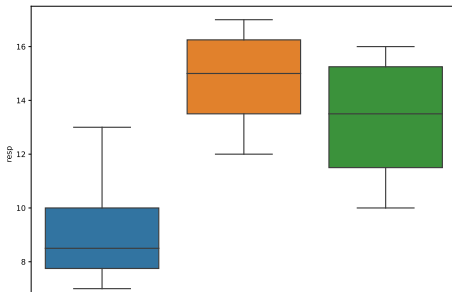
No Breakfast	Light Breakfast	Full Breakfast
8	14	10
7	16	12
9	12	16
13	17	15

- The response variable is attention span.
- The experimenter chooses 3 levels of a single factor - breakfast
- Each level of the factor is a treatment
- The experiment is replicated 4 times

# Design of Experiments

- Box plot

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
Breakfast=pd.DataFrame({"resp":[8,7,9,13,14,16,12,17,10,12,16,15],
'Trt':sum([["1"]*4, ["2"]*4, ["3"]*4],[[]])} )
# ["1"]*4+["2"]*4+["3"]*4
sns.boxplot(data=Breakfast, x="Trt", y='resp')
plt.show()
```



# Design of Experiments

- R code for Box-plot

```
resp=c(8,7,9,13,14,16,12,17,10,12,16,15)
Trt=c(rep(1,4),rep(2,4),rep(3,4))
Breakfast=as.data.frame(cbind(resp, Trt))
Breakfast$Trt =factor(Breakfast$Trt) #make "Trt" a factor
```

```
library(ggplot2)
ggplot(data=Breakfast, aes(x=Trt, y=resp,color=Trt))+
  geom_boxplot(alpha = 0.5) #set transparency
  geom_point()
```

# Design of Experiments

- Completely randomized design:
  - ▶ one factor only; one-way classification.
  - ▶ The treatments are assigned completely randomly to the experiment units.
- The data will be like

$Trt_1$	$Trt_2$	$\cdots$	$Trt_k$
$y_{11}$	$y_{21}$	$\cdots$	$y_{k1}$
$y_{12}$	$y_{22}$	$\cdots$	$y_{k2}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$
$y_{1,n_1}$	$y_{2,n_2}$	$\cdots$	$y_{k,n_k}$



# Design of Experiments

- Statistical model: By regarding each  $n_i$  observations as a random sample from an infinite population, the probability model (**means model**) is:

$$Y_{ij} = \mu_i + \varepsilon_{ij}, j = 1, 2, \dots, n_i, i = 1, 2, \dots, k,$$

where  $\varepsilon_{ij} \sim N(0, \sigma^2)$  are independent normal random errors with common variance  $\sigma^2$ .

- Are the  $k$  population means the same, or is at least one mean different from the others?
- We are testing  $H_0 : \mu_1 = \mu_2 = \dots = \mu_k$  versus  $H_a$  : At least two means are different from each other or “Not all means are equal”

# 1-way ANOVA

- **Effects model**

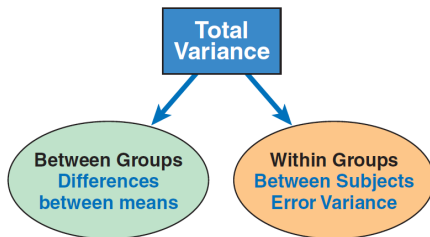
$$Y_{ij} = \mu + \tau_i + \varepsilon_{ij}, j = 1, 2, \dots, n_i, i = 1, 2, \dots, k,$$

- $\varepsilon_{ij} \sim N(0, \sigma^2)$  are independent normal random errors
- side condition:  $\sum_i \tau_i = 0$
- Model fit and factor effects
  - ▶ Overall mean:  $\hat{\mu} = \bar{y}_{..}$
  - ▶ Treatment:  $\hat{\tau}_i = \bar{y}_{i.} - \bar{y}_{..}$ ,  $i = 1, \dots, k$

# 1-way ANOVA

- The idea behind the Analysis of Variance(ANOVA) is to divide the total variation among the responses into the variance between groups, and that within groups, and see if those distributions match the null hypothesis that all groups come from the same distribution.

$$SS_{total} = SS_T + SS_E$$



Partitioning the total variance in a one-way ANOVA.

# 1-way ANOVA

$Trt_1$	$Trt_2$	$\cdots$	$Trt_k$	
$y_{11}$	$y_{21}$	$\cdots$	$y_{k1}$	
$y_{12}$	$y_{22}$	$\cdots$	$y_{k2}$	
$\vdots$	$\vdots$	$\vdots$	$\vdots$	
$y_{1,n_1}$	$y_{2,n_2}$	$\cdots$	$y_{k,n_k}$	
$\bar{y}_{1\cdot}$	$\bar{y}_{2\cdot}$	$\cdots$	$\bar{y}_{k\cdot}$	$\bar{y}_{\cdot\cdot}$

That is,

$$\sum \sum (y_{ij} - \bar{y}_{\cdot\cdot})^2 = \sum \sum (\bar{y}_{i\cdot} - \bar{y}_{\cdot\cdot})^2 + \sum \sum (y_{ij} - \bar{y}_{i\cdot})^2$$

- Degrees of freedom are additive, just like the sums of squares. Partition of the dfs:

$$N - 1 = (k - 1) + (N - k),$$

where  $N = \sum n_i$

# 1-way ANOVA

- A large value of  $SS_T$  reflects large differences in treatment means; A small value of  $SS_T$  likely indicates no differences in treatment means
- While sums of squares cannot be directly compared to test the hypothesis of equal means, mean squares can be compared.
- A mean square is a sum of squares divided by its degrees of freedom.

$$MS_T = SS_T / (k - 1), \quad MSE = SSE / (N - k).$$

- If the treatment means are equal, the treatment and error mean squares will be (theoretically) equal.
- If treatment means differ, the treatment mean square will be larger than the error mean square.

# 1-way ANOVA

- Remember that  $\sigma^2$  is the common variance for all a populations. The quantity  $MSE = SSE/(N - k)$  is a pooled estimate of  $\sigma^2$ , a weighted average of all a sample variances, whether or not  $H_0$  is true.
- If  $H_0$  is true, then the variation in the sample means, measured by  $MS_T = SS_T/(k - 1)$ , also provides an unbiased estimate of  $\sigma^2$ .
- However, if  $H_0$  is false and the population means are different, then  $MS_T$ —which measures the variance in the sample means — is unusually large. The test statistic

$$F = MS_T/MSE \sim F(df_1 = k - 1, df_2 = N - k) \text{ under } H_0$$

tends to be larger than usual.

- The F-test is a right-tailed statistical test.

# 1-way ANOVA

- 1-way ANOVA Table

**Table 1:** 1-way ANOVA Table

Source	df	SS	MS	F
Treatments	$k - 1$	$SS_T$	$SS_T / (k - 1)$	$MS_T / MSE$
Error	$N - k$	$SSE$	$SSE / (N - k)$	
Total	$N - 1$	$SS_{total}$		

- $E(MSE) = \sigma^2$ ,  $E(MS_T) = \sigma^2 + \frac{\sum_{i=1}^k n_i \tau_i^2}{k-1}$ .
- A completely randomized design is called balanced if the numbers of observations in all treatment are equal.
- Analysis of variance procedures are fairly robust against departures from normality when the design is balanced and when the data are fairly mound-shaped.

# 1-way ANOVA

- `f_oneway` function in `scipy.stats` takes the groups as input and returns ANOVA F and p value
  - ▶ [https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.f\\_oneway.html](https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.f_oneway.html)
- Fitting models using R-style formulas: Since version 0.5.0, **statsmodels** allows users to fit statistical models using R-style formulas.
  - ▶ <https://www.statsmodels.org/stable/examples/notebooks/generated/formulas.html>
  - ▶ [Patsy formula language description](https://patsy.readthedocs.io/en/latest/) <https://patsy.readthedocs.io/en/latest/>
- Getting data from R packages
  - ▶ mpg data from R package `ggplot2`

```
import statsmodels.api as sm
mpgclass=sm.datasets.get_rdataset("mpg", "ggplot2", cache=True)
mpg=mpgclass.data
mpg.head()
```

```
##   manufacturer model  displ  year  cyl      trans drv   cty   hwy fl    c
## 0          audi   a4     1.8  1999    4    auto(l5)  f    18    29 p  com
## 1          audi   a4     1.8  1999    4  manual(m5)  f    21    29 p  com
## 2          audi   a4     2.0  2008    4  manual(m6)  f    20    31 p  com
## 3          audi   a4     2.0  2008    4    auto(av)  f    21    30 p  com
## 4          audi   a4     2.8  1999    6    auto(l5)  f    16    26 p  com
```



# 1-way ANOVA

- I prefer to get ANOVA table as R like output using statsmodels
  - ▶ [https://www.statsmodels.org/dev/generated/statsmodels.regression.linear\\_model.OLS.html](https://www.statsmodels.org/dev/generated/statsmodels.regression.linear_model.OLS.html)
  - ▶ [https://www.statsmodels.org/dev/generated/statsmodels.regression.linear\\_model.OLS.fit.html](https://www.statsmodels.org/dev/generated/statsmodels.regression.linear_model.OLS.fit.html)

```
import statsmodels.api as sm
from statsmodels.formula.api import ols
Model1=ols('resp ~ Trt', data=Breakfast).fit(alpha=0.01)
anova_table = sm.stats.anova_lm(Model1, typ=2)
anova_table
```

##	sum_sq	df	F	PR(>F)
## Trt	64.666667	2.0	4.995708	0.03472
## Residual	58.250000	9.0	NaN	NaN

# 1-way ANOVA

```
print(Model1.summary())
```

```
##                                OLS Regression Results
## =====
## Dep. Variable:                 resp    R-squared:                 0.526
## Model:                        OLS      Adj. R-squared:            0.421
## Method:                       Least Squares    F-statistic:              4.996
## Date:                         Mon, 13 Feb 2023    Prob (F-statistic):      0.0347
## Time:                         18:13:22    Log-Likelihood:         -26.506
## No. Observations:             12    AIC:                    59.01
## Df Residuals:                 9    BIC:                    60.47
## Df Model:                     2
## Covariance Type:              nonrobust
## =====
##                coef      std err          t      P>|t|      [0.025      0.975]
## -----
## Intercept           9.2500        1.272        7.272      0.000        6.372      12.128
## Trt[T.2]            5.5000        1.799        3.057      0.014        1.431        9.569
## Trt[T.3]            4.0000        1.799        2.224      0.053       -0.069        8.069
## =====
## Omnibus:                1.660    Durbin-Watson:            2.361
## Prob(Omnibus):          0.436    Jarque-Bera (JB):         0.849
## Skew:                   0.167    Prob(JB):                 0.654
## Kurtosis:               1.740    Cond. No.                  3.73
## =====
##
```

# One-Way ANOVA

- Point estimation of the means  $\mu_1, \mu_2$  and  $\mu_3$

```
import numpy as np
Breakfast.groupby(['Trt']).agg(mean_resp=("resp", np.mean))
```

```
##      mean_resp
## Trt
## 1          9.25
## 2         14.75
## 3         13.25
```

```
np.mean(Breakfast['resp']) # estimate of  $\mu$ 
```

```
## 12.416666666666666
```

- In the last slide, we see
  - ▶ Intercept is estimate of  $\mu + \tau_1$
  - ▶ Trt[T.2] is estimate of  $\tau_2 - \tau_1$
  - ▶ Trt[T.3] is estimate of  $\tau_3 - \tau_1$

# One-Way ANOVA

- Interval estimation of the mean parameters
  - [https://www.statsmodels.org/dev/generated/statsmodels.regression.linear\\_model.OLSResults.conf\\_int.html](https://www.statsmodels.org/dev/generated/statsmodels.regression.linear_model.OLSResults.conf_int.html)

```
Model1.conf_int(alpha=0.05)
```

	0	1
## Intercept	6.372472	12.127528
## Trt[T.2]	1.430562	9.569438
## Trt[T.3]	-0.069438	8.069438

# 1-way ANOVA

## Comments:

- ANOVA's three types of estimating Sums of Squares: [https://md.psych.bio.uni-goettingen.de/mv/unit/lm\\_cat/lm\\_cat\\_unbal\\_ss\\_explained.html](https://md.psych.bio.uni-goettingen.de/mv/unit/lm_cat/lm_cat_unbal_ss_explained.html)
- ANOVA with Python: <https://www.statsmodels.org/stable/anova.html>
  - ▶ function OLS: [https://www.statsmodels.org/stable/generated/statsmodels.regression.linear\\_model.OLS.html](https://www.statsmodels.org/stable/generated/statsmodels.regression.linear_model.OLS.html)
  - ▶ anova\_lm [https://www.statsmodels.org/stable/generated/statsmodels.stats.anova.anova\\_lm.html](https://www.statsmodels.org/stable/generated/statsmodels.stats.anova.anova_lm.html)

# 1-way ANOVA

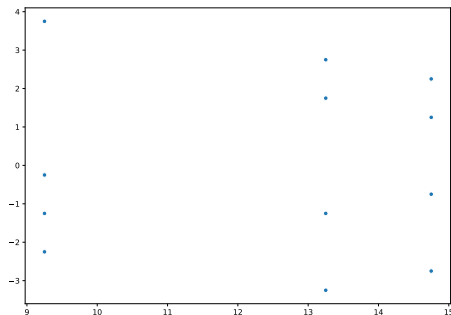
- R code

```
Model1=lm(resp ~ Trt, data= Breakfast)  
anova(Model1)  
summary(Model1)
```

# 1-way ANOVA

- Model adequacy check. We need to check the fitted values and residuals to see if the three model assumptions are satisfied.
- Residual plot to check independence and constant variance

```
import matplotlib.pyplot as plt  
sns.scatterplot(x=Model1.fittedvalues, y=Model1.resid)  
plt.show()
```



# 1-way ANOVA

- Levene test is equivalent to a 1-way ANOVA with the dependent variable being the absolute value of the difference between each observation  $y_{ij}$  and the mean/median of the group to which the observation  $y_{ij}$  belongs.
  - ▶ <https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.levene.html> which is a little lengthy to apply
- If data is drawn from normal distribution, we can use Bartlett's test to check the Homogeneity of variances.
  - ▶ <https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.bartlett.html>

```
from scipy.stats import levene
levene(Breakfast[Breakfast.Trt=='1'].resp,
Breakfast[Breakfast.Trt=='2'].resp,
Breakfast[Breakfast.Trt=='3'].resp)
```

```
## LeveneResult(statistic=0.18461538461538465, pvalue=0.834494695708
```



# 1-way ANOVA

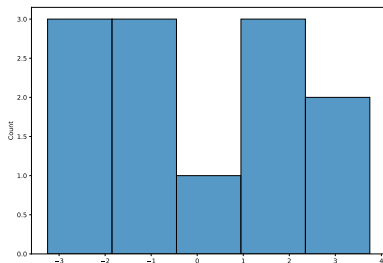
- R code

```
library(car)
Fitted= Model1$fitted
Resd= Model1$residuals
Breakfast=as.data.frame(cbind(Breakfast, Fitted, Resd))
leveneTest(Resd ~ Trt, data=Breakfast)
```

# 1-way ANOVA

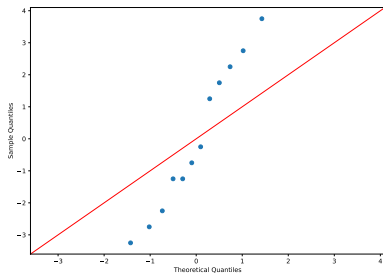
- Normality assumption can be checked using histogram and QQ plot of the residuals.

```
sns.histplot(Model1.resid)  
plt.show()
```



# 1-way ANOVA

```
sm.qqplot(Model1.resid, line='45')  
plt.show()
```



# 1-way ANOVA

- R code

```
par(mfrow=c(1,2))  
hist(Resd)  
box()  
qqnorm(Resd)  
qqline(Resd)
```

# 1-way ANOVA

- It's possible to use a significance test comparing the sample distribution to a normal one.
- Shapiro-Wilk's method is widely recommended for normality test. It is based on the correlation between the data and the corresponding normal scores (Ghasemi and Zahediasl 2012).

```
import scipy.stats as stats
stats.shapiro(Model1.resid)
```

```
## ShapiroResult(statistic=0.9501346349716187, pvalue=0.6389536261558533)
```

- R code

```
shapiro.test(Resd)
```

## 2-Way ANOVA for Block Design

- The Blocking Principle
  - ▶ Blocking is a technique for dealing with nuisance factors
  - ▶ A nuisance factor is a factor that probably has some effect on the response, but it's of no interest to the experimenter. . . however, the variability it transmits to the response needs to be minimized
- Typical nuisance factors include batches of raw material, operators, pieces of test equipment, time (shifts, days, etc.), different experimental units

# 2-Way ANOVA for Block Design

## Randomized Complete Block Design

- There are two factors: factor of interest and nuisance factor.
- A block is a specific level of the nuisance factor
- The design uses blocks such that units in each block are relatively similar or homogeneous, with one unit within each block randomly assigned to each treatment (all runs within a block are randomized).
- If the design involves  $a$  treatments within each of  $b$  blocks, then the total number of observations is  $ab$ .
- Variability between blocks can be large, variability within a block should be relatively small
- The purpose of blocking is to isolate the block-to-block variability that might hide/affect the effect of the treatments.
- The design must be balanced: the number of units in each block must be equal to the number of treatments.

## 2-Way ANOVA for Block Design

- Example: We want to investigate the effect of 3 methods of soil preparation on the growth of seedlings. Each method is applied to seedlings growing at each of 4 locations and the average first year growth is recorded.

**Table 2:** Seedling Problem

Soil Prep	Location			
	1	2	3	4
A	11	13	16	10
B	15	17	20	12
C	10	15	13	10

- Treatment = soil preparation ( $a = 3$ )
- Block = location ( $b = 4$ )
- Is the average growth different for the 3 soil preps?



## 2-Way ANOVA for Block Design

- Suppose that there are  $a$  treatments (factor levels) and  $b$  blocks
- Let  $y_{ij}$  be the response for the  $i$ -th treatment applied to the  $j$ -th block.  
 $i = 1, 2, \dots, a, j = 1, 2, \dots, b$
- The data will be look like this

**Table 3:** Data from Block Design

Blocks	Treatments			
	1	2	...	$a$
1	$y_{11}$	$y_{21}$	...	$y_{a1}$
2	$y_{12}$	$y_{22}$	...	$y_{a2}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$b$	$y_{1b}$	$y_{2b}$	13	$y_{ab}$

## 2-Way ANOVA for Block Design

- Statistical model:

$$Y_{ij} = \mu + \tau_i + \beta_j + \varepsilon_{ij}, j = 1, 2, \dots, b, i = 1, 2, \dots, a,$$

where  $\varepsilon_{ij} \sim N(0, \sigma^2)$  are independent normal random errors with common variance  $\sigma^2$ .

- The relevant null hypothesis for comparing the  $a$  treatment means

$$H_0 : \mu_1 = \mu_2 = \dots = \mu_a$$

## 2-Way ANOVA for Block Design

- Partition of the total variation:

$$SS_{total} = SS_T + SS_B + SS_E$$

That is,

$$\begin{aligned} \sum \sum (y_{ij} - \bar{y}_{..})^2 &= \sum \sum (\bar{y}_{i.} - \bar{y}_{..})^2 + \sum \sum (\bar{y}_{.j} - \bar{y}_{..})^2 \\ &\quad + \sum \sum (y_{ij} - \bar{y}_{i.} - \bar{y}_{.j} + \bar{y}_{..})^2, \end{aligned}$$

where  $y_{i.} = \sum_{j=1}^b y_{ij}, i = 1, \dots, a; y_{.j} = \sum_{i=1}^a y_{ij}, j = 1, \dots, b.$

- The corresponding degrees of freedom for the sums of squares are

$$ab - 1 = (a - 1) + (b - 1) + (a - 1)(b - 1).$$

- Therefore, ratios of sums of squares to their degrees of freedom result in mean squares and the ratio of the mean square for treatments to the error mean square is an  $F$  statistic that can be used to test the hypothesis of equal treatment means.

## 2-Way ANOVA for Block Design

- 2-way ANOVA Table

**Table 4:** 2-way ANOVA Table for Block Design

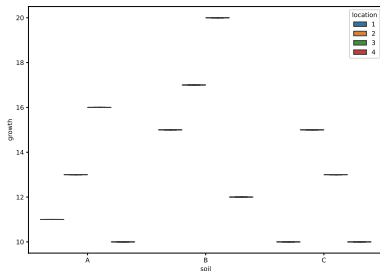
Source	df	SS	MS	F
Treatments	$a - 1$	$SS_T$	$SS_T / (a - 1)$	$MS_T / MSE$
Blocks	$b - 1$	$SS_B$	$SS_B / (b - 1)$	$MS_B / MSE$
Error	$(a - 1)(b - 1)$	$SSE$	$SSE / (a - 1)(b - 1)$	
Total	$ab - 1$	$SS_{total}$		

- Larger values of the F-test statistic (resulting in smaller p-values), so the ANOVA test is right-tailed.
- If the  $b$  block means are not significantly different, the  $b$  blocks are homogeneous and the block design is not necessary for future same study.

## 2-Way ANOVA for Block Design

- The Seedling Problem

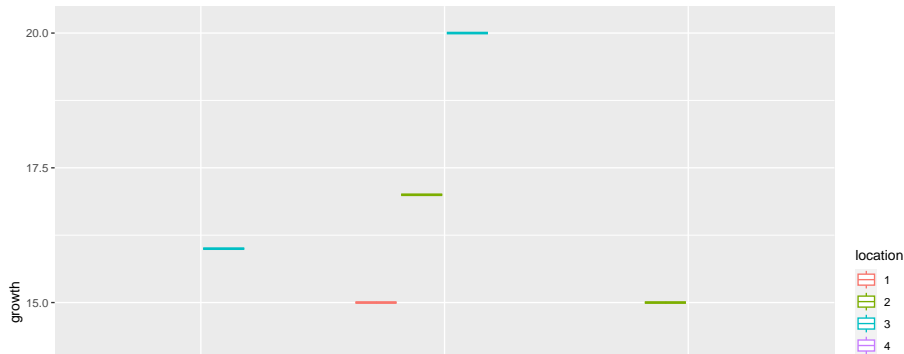
```
import pandas as pd
import matplotlib.pyplot as plt
seedlings=pd.read_csv("../data/Soil.csv")
import seaborn as sns
sns.boxplot(x="soil", y="growth", hue="location", data=seedlings)
plt.show()
```



# 2-Way ANOVA for Block Design

- R code

```
library(readr)
seedlings = read_csv("../data/Soil.csv")
seedlings$location=as.factor(seedlings$location)
seedlings$soil=as.factor(seedlings$soil)
library(ggplot2)
ggplot(data=seedlings,aes(x=soil,y=growth,color=location))+
  geom_boxplot()
```



## 2-Way ANOVA for Block Design

- We use the same function `ols`

```
import statsmodels.api as sm
from statsmodels.formula.api import ols
Model2=ols('growth ~ soil+location', data=seedlings).fit()
anova_table = sm.stats.anova_lm(Model2, typ=2)
print(anova_table)
```

##	sum_sq	df	F	PR(>F)
## soil	38.000000	2.0	2.113068	0.183317
## location	1.066667	1.0	0.118628	0.739406
## Residual	71.933333	8.0	NaN	NaN

## 2-Way ANOVA for Block Design

```
print(Model2.summary())
```

```
##                                OLS Regression Results
## =====
## Dep. Variable:                growth    R-squared:
## Model:                        OLS       Adj. R-squared:
## Method:                      Least Squares    F-statistic:
## Date:                        Mon, 13 Feb 2023    Prob (F-statistic):
## Time:                        18:13:49          Log-Likelihood:
## No. Observations:            12              AIC:
## Df Residuals:                8              BIC:
## Df Model:                    3
## Covariance Type:            nonrobust
## =====
##                                coef      std err          t      P>|t|      [0.025
## -----
## Intercept                   13.1667       2.448       5.378     0.001     7.521
## soil[T.B]                   3.5000       2.120       1.651     0.137    -1.390
## soil[T.C]                   -0.5000       2.120      -0.236     0.820    -5.390
## location                    -0.2667       0.774      -0.344     0.739    -2.052
## =====
## Omnibus:                     1.599    Durbin-Watson:
```



## 2-Way ANOVA for Block Design

- R code

```
Model2 <- lm(growth ~ location +soil, data=seedlings)
anova(Model2)
summary(Model2)
```

## 2-Way ANOVA for Block Design

- Model adequacy check: We need to check the fitted values and residuals to see if the three model assumptions are satisfied. These are to be done in a similar way as those in one-way ANOVA.
- Residual plot

```
import matplotlib.pyplot as plt
sns.scatterplot(x=Model2.fittedvalues, y=Model2.resid)
plt.show()
```

- R code

```
Fitted=Model2$fitted
Resd=Model2$residuals
seedlings =as.data.frame(cbind(seedlings, Fitted, Resd))
plot(Resd~ Fitted, data= seedlings)
```

## 2-Way ANOVA for Block Design

- Now we use Levene test to further check the constant variance assumptions. That is, we are testing

$$H_0 : \sigma_A = \sigma_B = \sigma_C$$

$$H_0 : \sigma_1 = \sigma_2 = \sigma_3 = \sigma_4$$

- We fail to reject both null hypotheses because the p-values of both tests are large
  - The Levene's test `.levne()` in library `bioinfokit` has R-like syntax

```
from scipy.stats import levene
levne(seedlings[seedlings.soil=='A'].growth,
seedlings[seedlings.soil=='B'].growth,
seedlings[seedlings.soil=='C'].growth)
```

```
## LeveneResult(statistic=0.16666666666666669, pvalue=0.849035398860626)
```

```
levne(seedlings[seedlings.location==1].growth,
seedlings[seedlings.location==2].growth,
seedlings[seedlings.location==3].growth,
seedlings[seedlings.location==4].growth)
```

```
## LeveneResult(statistic=0.5098039215686275, pvalue=0.6865570881401991)
```

## 2-Way ANOVA for Block Design

- R code

```
library(car)
seedlings$soil = as.factor(seedlings$soil)
seedlings$location = as.factor(seedlings$location)
leveneTest(Resd ~ soil, data=seedlings)
leveneTest(Resd ~ location, data=seedlings)
```

## 2-Way ANOVA for Block Design

- Normality assumption can be checked using histogram and QQ plot of the residuals.

```
sns.histplot(Model2.resid)
plt.show()
sm.qqplot(Model2.resid, line='45')
plt.show()
```

- R code

```
par(mfrow=c(1,2))
hist(Resd)
box()
qqnorm(Resd)
qqline(Resd)
```

## 2-Way ANOVA for Block Design

- Normality test

```
import scipy.stats as stats
stats.shapiro(Model2.resid)
```

```
## ShapiroResult(statistic=0.9275646209716797, pvalue=0.355012297630)
```

- R code

```
shapiro.test(Resd)
```

# 2-Way ANOVA for Factorial Design

## Two-factor Factorial Design

- A two-way classification which involves two factors, both of which are of interest to the experimenter.
- There are  $a$  levels of factor A and  $b$  levels of factor B — the experiment is replicated  $n$  times at each factor-level combination.
- The effect of a factor is defined to be the change in mean response produced by a change in the level of the factor (generally when the factor is changed from low to high)
- The replications in the design allow the experimenter to investigate the interaction between factors A and B.

# 2-Way ANOVA for Factorial Design

## Interaction

- There is an interaction between two factors if the effect of one of the factors changes for different levels of the other factor.
- Interaction describes the effect of one factor on the behavior of the other. If there is no interaction, the two factors behave independently.
- Example: The data in the table are categorized with two factors:
  - ▶ Sex: Male or Female
  - ▶ Blood Lead Level: Low, Medium, or High
- The subcategories are called cells, and the response variable is IQ score. The data are presented on the next slide:



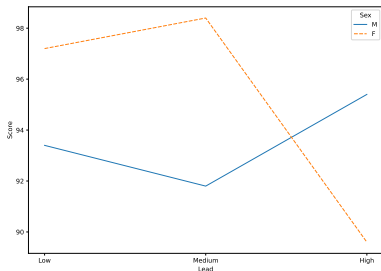
## 2-Way ANOVA for Factorial Design

Measures of Performance IQ			
	Blood Lead Level		
	Low	Medium	High
Male	85	78	93
	90	107	97
	107	90	79
	85	83	97
	100	101	111
Female	64	97	100
	111	80	71
	76	108	99
	136	110	85
	99	97	93

## 2-Way ANOVA for Factorial Design

- Let's explore the IQ data in the table by calculating the mean for each cell and constructing an interaction graph.

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
IQ = pd.read_csv("../data/IQPerformance.csv")
sns.lineplot(data=IQ, x="Lead", y="Score", hue="Sex", style="Sex",
ci= None) #ci=None removes the confidence band
plt.show()
```



## 2-Way ANOVA for Factorial Design

- R code

```
library(readr)
library(ggplot2)
IQ = read_csv("../data/IQPerformance.csv")
IQ$Lead=as.factor(IQ$Lead)
IQ$Sex=as.factor(IQ$Sex)
ggplot(IQ, aes(x=Lead, y=Score, group=Sex, col=Sex)) +
  geom_point() + stat_summary(fun= mean, geom = "line")
#with(IQ, interaction.plot(x.factor = Lead,
#                          trace.factor = Sex, response = Score))
```

# 2-Way ANOVA for Factorial Design

## Interaction

- An interaction effect is suggested if the line segments are far from being parallel.
- No interaction effect is suggested if the line segments are approximately parallel.
- For the IQ scores, it appears there is an interaction effect:
  - ▶ Females with high lead exposure appear to have lower IQ scores, while males with high lead exposure appear to have high IQ scores.

## 2-Way ANOVA for Factorial Design

- Let  $y_{ijk}$  be the  $k$ -th replication at the  $i$ -th level of A and the  $j$ -th level of B.  $i = 1, 2, \dots, a, j = 1, 2, \dots, b, k = 1, 2, \dots, r$ .
- The data will be look like this

**Table 5:** Data from a Factorial Design

Factor B	Factor A			
	1	2	...	$a$
1	$y_{111}, \dots, y_{11r}$	$y_{211}, \dots, y_{21r}$	...	$y_{a11}, \dots, y_{a1r}$
2	$y_{121}, \dots, y_{12r}$	$y_{221}, \dots, y_{22r}$	...	$y_{a21}, \dots, y_{a2r}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$b$	$y_{1b1}, \dots, y_{1br}$	$y_{2b1}, \dots, y_{2br}$	...	$y_{ab1}, \dots, y_{abr}$

## 2-Way ANOVA for Factorial Design

- Statistical model:

$$Y_{ijk} = \mu + \tau_i + \beta_j + (\tau\beta)_{ij} + \varepsilon_{ijk}, k = 1, 2, \dots, n, j = 1, 2, \dots, b, i = 1, 2, \dots, a,$$

where  $\varepsilon_{ijk} \sim N(0, \sigma^2)$  are independent normal random errors with common variance  $\sigma^2$ .

- The  $SS_{total}$  is divided into 4 parts:
  - ▶ SSA (sum of squares for factor A): measures the variation among the means for factor A
  - ▶ SSB (sum of squares for factor B): measures the variation among the means for factor B
  - ▶ SS(AB) (sum of squares for interaction): measures the variation among the ab combinations of factor levels
  - ▶ SSE (sum of squares for error): measures experimental error

## 2-Way ANOVA for Factorial Design

- Partition of the total variation:

$$SS_{total} = SSA + SSB + SS(AB) + SS_E$$

That is,

$$\begin{aligned}\sum \sum \sum (y_{ijk} - \bar{y}_{...})^2 &= \sum \sum \sum (\bar{y}_{i..} - \bar{y}_{...})^2 + \sum \sum \sum (\bar{y}_{.j.} - \bar{y}_{...})^2 \\ &\quad + \sum \sum \sum (\bar{y}_{ij.} - \bar{y}_{i..} - \bar{y}_{.j.} + \bar{y}_{...})^2 \\ &\quad + \sum \sum \sum (y_{ijk} - \bar{y}_{ij.})^2.\end{aligned}$$

- The corresponding degrees of freedom for the sums of squares are

$$abr - 1 = (a - 1) + (b - 1) + (a - 1)(b - 1) + ab(r - 1).$$

## 2-Way ANOVA for Factorial Design

- 2-way ANOVA Table

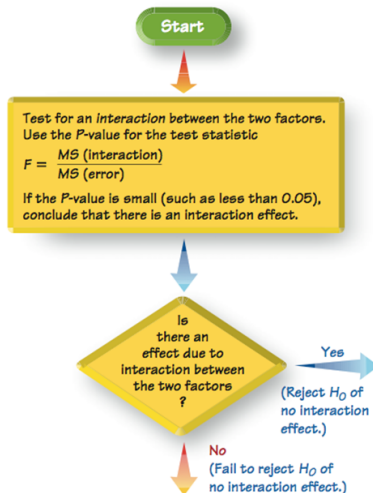
**Table 6:** 2-way ANOVA Table for Factorial Design

Source	df	SS	MS	F
Trt A	$a - 1$	$SSA$	$SSA/(a - 1)$	$MSA/MSE$
Trt B	$b - 1$	$SSB$	$SSB/(b - 1)$	$MSB/MSE$
Interaction	$(a - 1)(b - 1)$	$SS(AB)$	$SS(AB)/(a - 1)(b - 1)$	$MS(AB)/MSE$
Error	$ab(r - 1)$	$SSE$	$SSE/ab(r - 1)$	
Total	$abr - 1$	$SS_{total}$		

- Again, all ANOVA F-tests are right-tailed.
- Procedure for Two-Way ANOVA, see the next two slides



# 2-Way ANOVA for Factorial Design



## Procedure for Two-Way ANOVA

If the interaction is significant, the main effects are NOT tested, and we focus on the **differences** in the ab factor-level means. We look at the **interaction plot** to see where the differences lie

- ❖ Be sure to test for an **interaction** between the two factors first.

# 2-Way ANOVA for Factorial Design



No  
(Fail to reject  $H_0$  of  
no interaction effect.)

## Procedure for Two-Way ANOVA

Test for effect from row factor using the  $P$ -value for the test statistic

$$F = \frac{MS(\text{row factor})}{MS(\text{error})}$$

If the  $P$ -value is small (such as less than 0.05), conclude that there is an effect from the row factor.



Test for effect from column factor using the  $P$ -value for the test statistic

$$F = \frac{MS(\text{column factor})}{MS(\text{error})}$$

If the  $P$ -value is small (such as less than 0.05), conclude that there is an effect from the column factor.

If the interaction effect is not significant, the main effects A and B can be individually tested using  $F = MSA/MSE$  and  $F = MSB/MSE$ , respectively.

## 2-Way ANOVA for Factorial Design

- We use the same function `ols`

```
import statsmodels.api as sm
from statsmodels.formula.api import ols
Model3=ols('Score ~ Lead*Sex', data=IQ).fit()
anova_table = sm.stats.anova_lm(Model3, typ=2)
print(anova_table)
```

##	sum_sq	df	F	PR(>F)
## Lead	48.800000	2.0	0.099484	0.905676
## Sex	17.633333	1.0	0.071895	0.790889
## Lead:Sex	211.466667	2.0	0.431095	0.654730
## Residual	5886.400000	24.0	NaN	NaN

## 2-Way ANOVA for Factorial Design

```
print(Model3.summary())
```

```
##                                OLS Regression Results
## =====
## Dep. Variable:                Score    R-squared:
## Model:                      OLS        Adj. R-squared:
## Method:                     Least Squares    F-statistic:
## Date:                       Mon, 13 Feb 2023    Prob (F-statistic):
## Time:                       18:14:02          Log-Likelihood:
## No. Observations:           30              AIC:
## Df Residuals:               24              BIC:
## Df Model:                   5
## Covariance Type:            nonrobust
## =====
##                                coef      std err          t      P>|t|
## -----
## Intercept                   89.6000      7.004      12.793      0.000
## Lead[T.Low]                 7.6000      9.905       0.767      0.450
## Lead[T.Medium]              8.8000      9.905       0.888      0.383
## Sex[T.M]                    5.8000      9.905       0.586      0.564
## Lead[T.Low]:Sex[T.M]        -9.6000     14.008      -0.685      0.500
## Lead[T.Medium]:Sex[T.M]    -12.4000    14.008      -0.885      0.385
```

## 2-Way ANOVA for Factorial Design

- R code

```
fit3 = lm(Score ~ Lead +Sex +Lead*Sex, data=IQ)
#or lm(Score ~ Lead*Sex, data=IQ)
anova(fit3)
summary(fit3)
```

## 2-Way ANOVA for Factorial Design

- Model adequacy check: DIY
- R code

```
Fitted=fit3$fitted
Resd=fit3$residuals
IQ =as.data.frame(cbind(IQ, Fitted, Resd))
plot(Resd~ Fitted, data= IQ)
leveneTest(Resd ~ Lead, data=IQ)
leveneTest(Resd ~ Sex, data=IQ)
hist(Resd)
qqnorm(Resd)
qqline(Resd)
shapiro.test(Resd)
```

# Three-Way ANOVA

With more than one factors, it is recommended to use statistical modelling. However, as always with the analysis of statistical data, one should first inspect the data visually.

# License

This work is licensed under a [Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License](#).