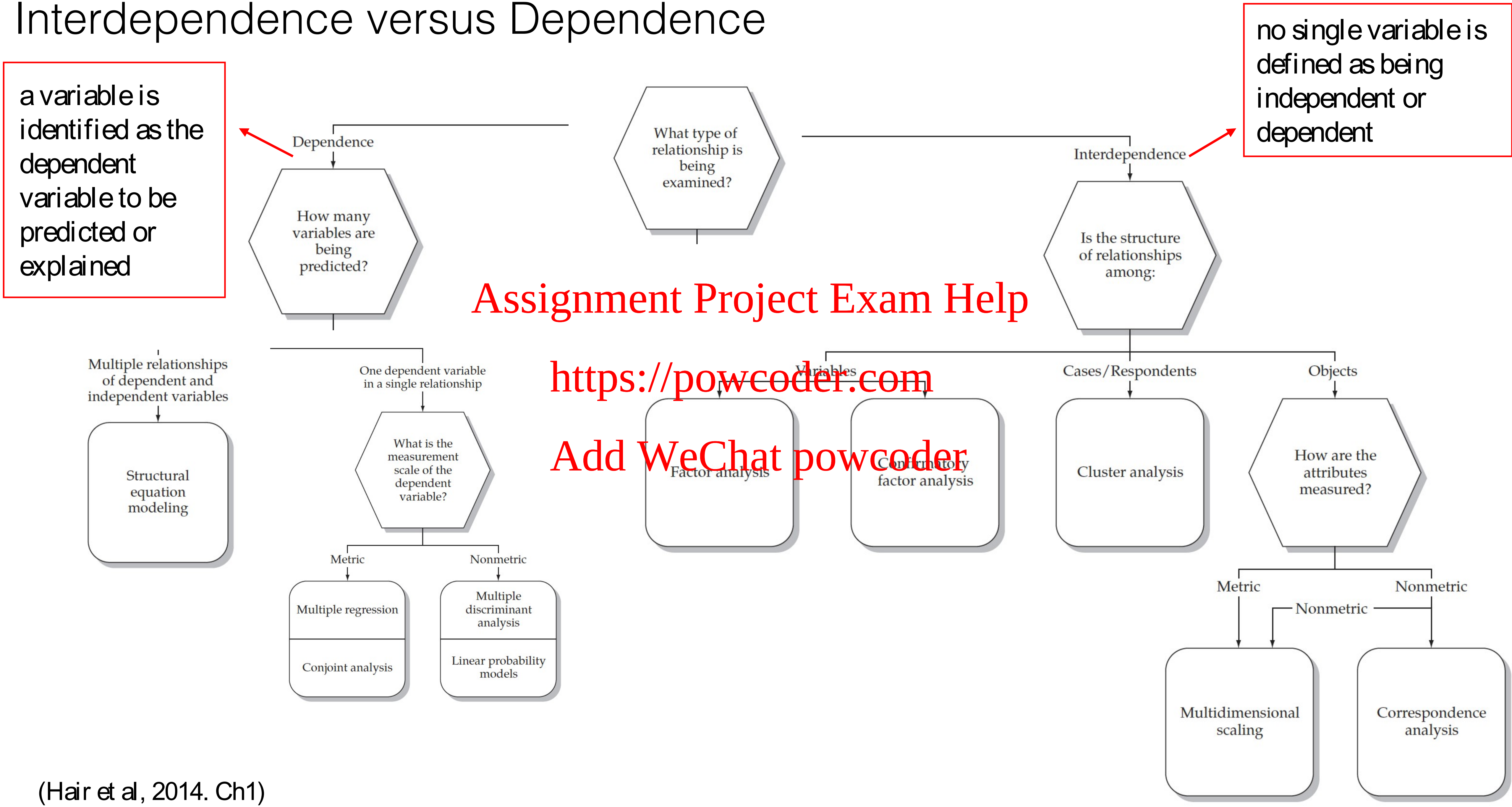
Factor Analysis

Ch.3 Multivariate Data Analysis. Joseph Hair et al. 2014. Pearson
Avilash Navlani. 2019. Introduction to Factor Analysis in Python
Jay Narayan. 2019. Multiple Linear Regression & Factor Analysis in R

Interdependence versus Dependence



(Hair et al, 2014. Ch1)

Dimensions

Data Matrix M

Each row - an observation in the space (the graph) also called sample
Each column - an attribute, also called dimension

```
np.array(data).shape
```

```
(150, 4)
```

```
array([[5.1, 3.5, 1.4, 0.2],
       [4.9, 3. , 1.4, 0.2],
       [4.7, 3.2, 1.3, 0.2],
       [4.6, 3.1, 1.5, 0.2],
       [5. , 3.6, 1.4, 0.2],
       [5.4, 3.9, 1.7, 0.4],
       [4.6, 3.4, 1.4, 0.3],
       [5. , 3.4, 1.5, 0.2],
       [4.4, 2.9, 1.4, 0.2],
       [4.9, 3.1, 1.5, 0.1],
       [5.4, 3.7, 1.5, 0.2],
```


150 observations (samples) and 4 dimensions

Overfitting

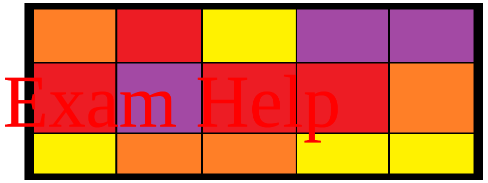
Irrelevant and correlated attributes can even decrease the performance in some algorithms

Factor analysis and PCA play a role in the reduction of these dimensions

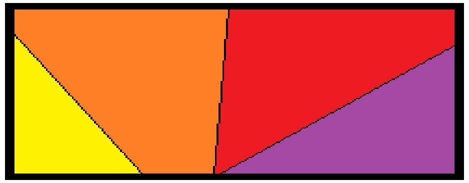
Principal Component and Factor Analysis

1. Statistical approaches to analyze interrelationships among a large number of variables and to explain these variables in terms of their common underlying dimensions (factors).

2. The same steps — extraction, interpretation, rotation, and choosing the number of factors or components.



FA : Find common variance and allign them as factors. All yellows, reds, oranges and purples form one factor each



PCA : The total variance is divided to form as many PCs as desired

3. Factor analysis makes assumptions and PCA does not. The basic assumption is that there are implicit features responsible for the features of the dataset

4. FA: we infer the existence of latent variables that explain the pattern of correlations among our observed variables

Avinash Navlani. 2019. Introduction to Factor Analysis.

FA Example

"*What underlying attitudes lead people to respond to the questions on a political survey?*

Examining the correlations among the survey items reveals that there is significant overlap among various subgroups of items--questions about taxes tend to correlate with each other, questions about military issues correlate with each other, and so on.

With factor analysis, you can investigate the number of underlying factors. Additionally, you can compute factor scores for each respondent, which can then be used in subsequent analyses. For example, you might build a logistic regression model to predict voting behavior based on factor scores."

IBM Knowledge Center

Factor Analysis

Univariate Techniques a single variable

Multivariate Techniques a possible correlation between many variables

How to manage these variables?

Factor Analysis

- Examines the interrelationships among a large number of variables and attempts to explain them in terms of their **common underlying dimensions**

factors

- A data reduction technique that does not have dependent and independent variables.

Terminology

Variance

How far the data is spread out

Unique Variance

Variance of the variable is not associated with other variables

Shared Variance

Variance is shared with other variances, creating redundancy in the data

Variate

the linear composite of variables

Example

Original Correlation Matrix
(no visible patterns)

	V_1	V_2	V_3	V_4	V_5	V_6	V_7	V_8	V_9
V_1 Price Level	1.00								
V_2 Store Personnel	.427	1.00							
V_3 Return Policy	.302	.771	1.00						
V_4 Product Availability	.470	.497	.427	1.00					
V_5 Product Quality	.765	.406	.307	.472	1.00				
V_6 Assortment Depth	.281	.445	.423	.713	.325	1.00			
V_7 Assortment Width	.354	.490	.471	.719	.378	.724	1.00		
V_8 In-Store Service	.242	.719	.733	.428	.240	.311	.435	1.00	
V_9 Store Atmosphere	.372	.737	.774	.479	.326	.429	.466	.710	1.00

(Hair et al. 2015. Ch3)

Correlation Matrix in Factor Analysis (three patterns)

	V_3	V_8	V_9	V_2	V_6	V_7	V_4	V_1	V_5
V_3 Return Policy	1.00								
V_8 In-store Service	.733	1.00							
V_9 Store Atmosphere	.774	.710	1.00						
V_2 Store Personnel	.741	.719	.787	1.00					
V_6 Assortment Depth	.423	.311	.429	.445	1.00				
V_7 Assortment Width	.471	.435	.468	.490	.724	1.00			
V_4 Product Availability	.427	.428	.479	.497	.713	.719	1.00		
V_1 Price Level	.302	.242	.372	.427	.281	.354	.470	1. 00	
V_5 Product Quality	.307	.240	.326	.406	.325	.378	.472	.765	1.00

(Hair et al, 2014. Ch3)

Factor 1: in-store experience

Factor 2: product offerings

Factor 3: value

Goal: **Grouping** highly **intercorrelated** variables into distinct sets (**factors**)

Usage: Market research, advertising, finance, operation research etc. (to identify brand features, channel selection criteria…)

Factor Analysis Outcomes

1. **Data summarization** = derives underlying dimensions that describe the data in a much smaller number of concepts than the original individual variables.

2. **Data reduction** = extends the process of data summarization by deriving an empirical value (factor score) for each dimension (factor) and then substituting this value for the original values.

The goal of data summarization is achieved by defining a small number of factors that adequately represent the original set of variables

The goal is to retain the **nature and character** of the original variables, but reduce their number to simplify the subsequent multivariate analysis

Types of Factor Analysis

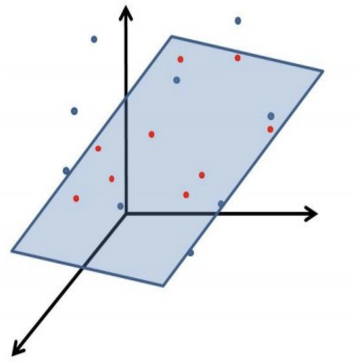
1. **Exploratory Factor Analysis EFA** = is used to discover the factor structure of a construct and examine its reliability. It is data driven.

2. **Confirmatory Factor Analysis CFA** = is used to confirm the fit of the hypothesized factor structure to the observed (sample) data. It is theory driven.

Factor Analysis is **interdependent technique** – no distinction between dependent and independent variables

Factor Analysis

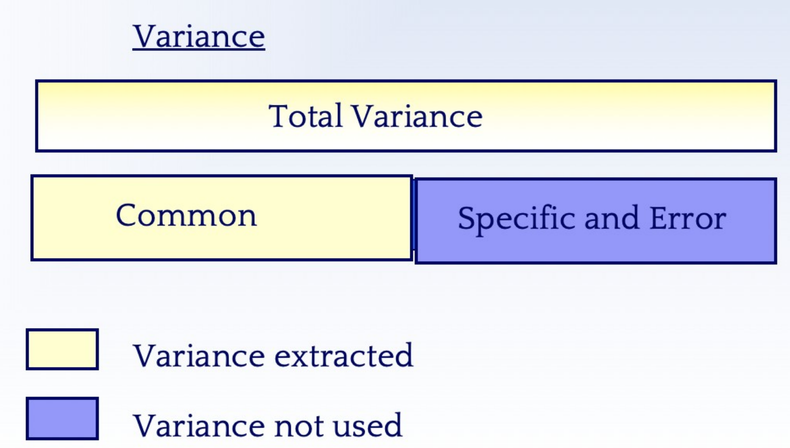
Each observable variable is a linear function of independent factors and error term



loadings independent factors error term

$$Y_i = \beta_{i0} + \beta_{i1}F_1 + \beta_{i2}F_2 + (1)e_i$$

$$Var(Y_i) = \underbrace{\beta_{i1}^2 + \beta_{i2}^2}_{\text{communality}} + \underbrace{\sigma_i^2}_{\text{specific variance}}$$



(Hair et al, 2014. Ch3)

The communality of the variable is the part that is explained by the common factors F1 and F2

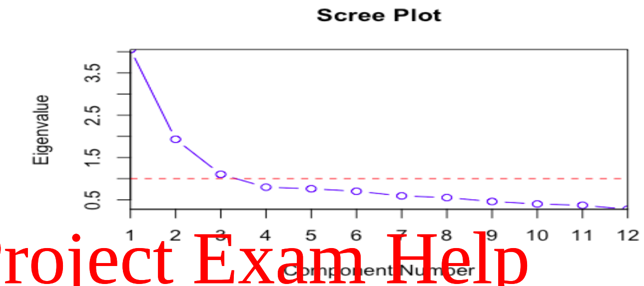
The specific variance is the part of the variance of Yi that is not accounted by the common factors

Loadings are the weights that the variable has for constructing a factor. The higher the load is, the more relevant in defining the factor's dimensionality.

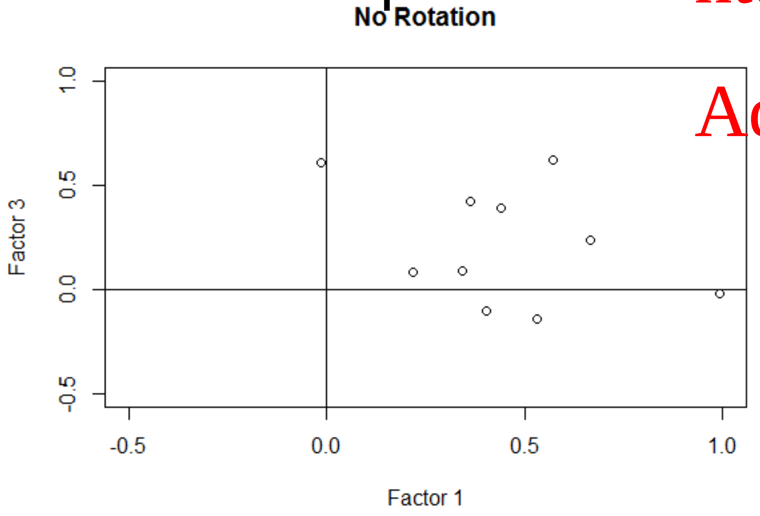
(Barbara Engelhart. 2013. Factor Analysis Lecture; Peter Tryfos. 1997. Chapter 14. Factor Ana;lysis)

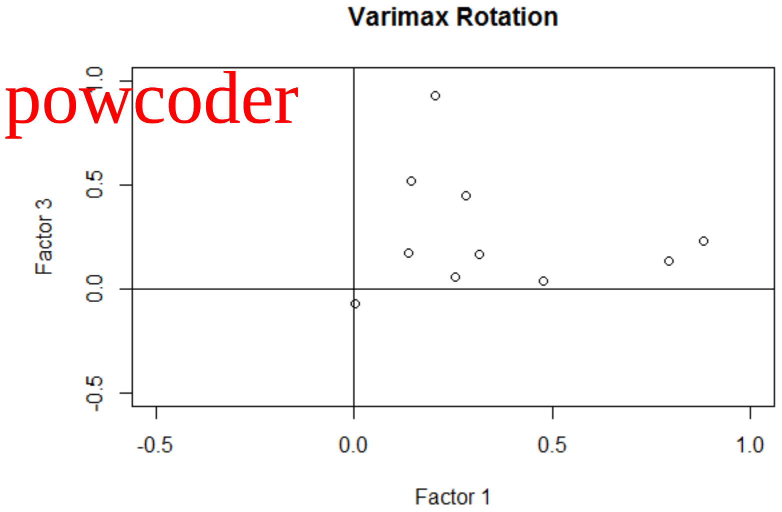
Two Steps

Factor Extraction Determine the number of factors: eigenvalue > 1 or "elbow" (Scree plot)



Factor Rotation The axes of the factors can be rotated within the multidimensional variable space





Varimax Method minimizes the number of variables that have high loadings on each factor.

Factor Analysis in R

Step 1 Data

```
library(readr)
library(psych)
library(tidyverse)
library(Hmisc)
library(car)
data <- read_csv("Factor-Hair-Revised.csv")
```

How many dimensions? dim(data) 100 x 13

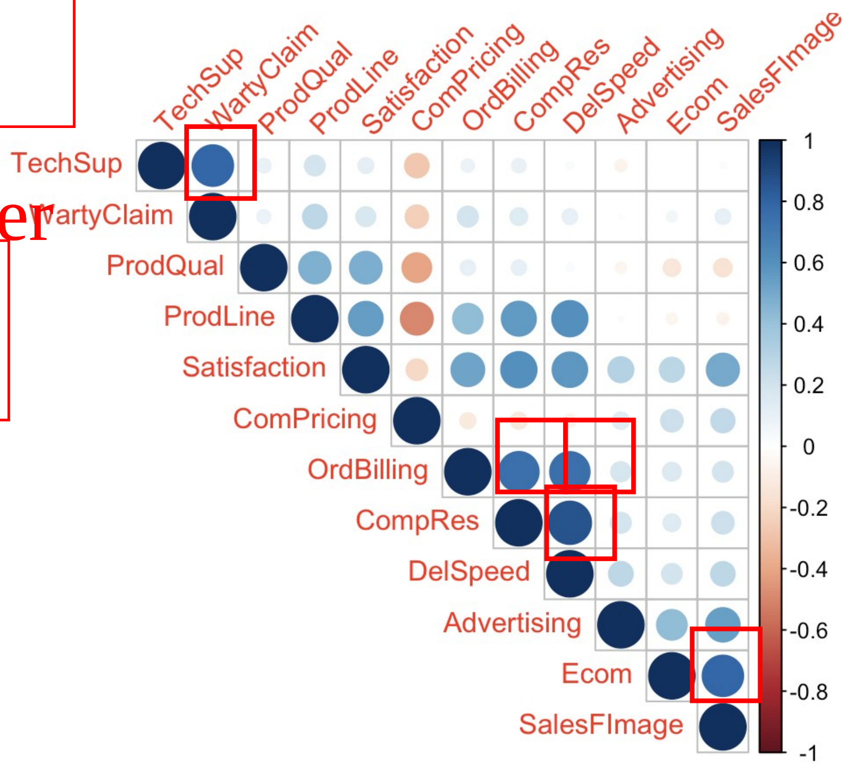
What are the variables types? str(data) all numeric except ID

What are the variable names? names(data)

Remove ID data_X <- select(data,-c(1))

Step 2 Correlation Matrix

```
datamatrix <- cor(data_X)
corrplot(datamatrix, order="hclust", type='upper', tl.srt = 45)
```



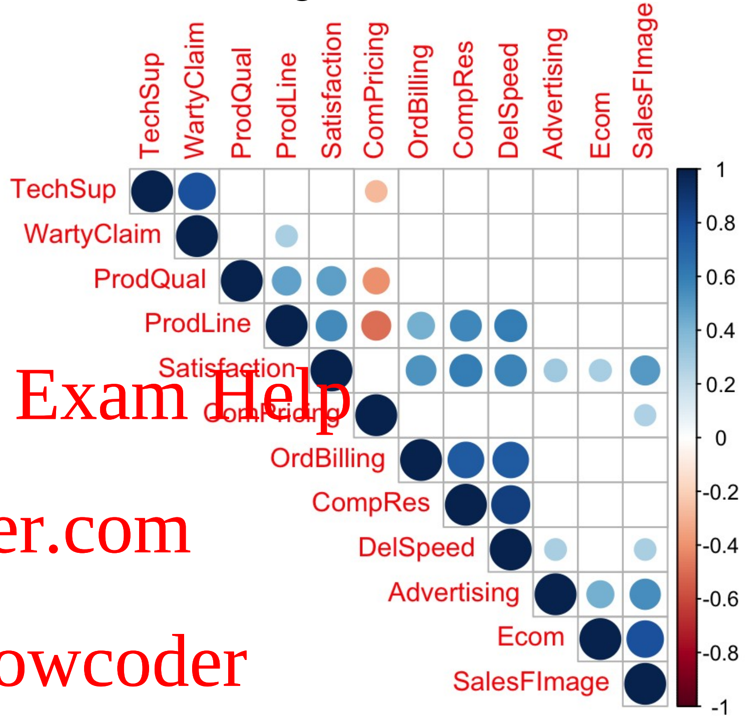
1. CompRes and DelSpeed are highly correlated
2. OrdBilling and CompRes are highly correlated
3. WartyClaim and TechSupport are highly correlated
4. OrdBilling and DelSpeed are highly correlated
5. Ecom and SalesFImage are highly correlated

Factor Analysis in R

Step 2 (cont.)

```
res2 <- rcorr(as.matrix(data_X), type="pearson")
# Extract the correlation coefficients
res2$r
# Extract p-values
res2$P
# Insignificant correlations are leaved blank
corrplot(res2$r, type="upper", order="hclust",
         p.mat = res2$P, sig.level = 0.01, insig = "blank")
```


Significant Pairs



Recall Assumptions of Linear Regression: Linearity, Homoscedasticity, Residuals normality, No Multicollinearity

```
model <- lm(Satisfaction ~., data = data_X)
vif(model)
```

VIF (High Variable Inflation Factor) > 2.5

```
   ProdQual        Ecom     TechSup     CompRes Advertising    ProdLine
   1.635797    2.756694    2.976796    4.730448    1.508933    3.488185
SalesFImage  ComPricing  WartyClaim  OrdBilling    DelSpeed
   3.439420    1.635000    3.198337    2.902999    6.516014
```

http://www.sthda.com/english/wiki/correlation-matrix-a-quick-start-guide-to-analyze-format-and-visualize-a-correlation-matrix-using-r-software

Factor Analysis in R

Step 3 Testing for FA - Kaiser-Meyer-Olkin (KMO)

- Test measures the suitability of data for factor analysis
- KMO values range between 0 and 1
- Value of KMO less than 0.6 is considered inadequate


```
data_fa <- data_X[,-12]
datamatrix <- cor(data_fa)
KMO(r=datamatrix)
```

Remove Dependent variable - Satisfaction


```
Kaiser-Meyer-Olkin factor adequacy
Call: KMO(r = datamatrix)
Overall MSA =  0.65
MSA for each item =
   ProdQual        Ecom     TechSup     CompRes Advertising    ProdLine
       0.51        0.63        0.52        0.79        0.78        0.62
SalesFImage  ComPricing  WartyClaim  OrdBilling    DelSpeed
       0.62        0.75        0.51        0.76        0.67
```

MSA > 0.5
Factor Analysis is appropriate on this data

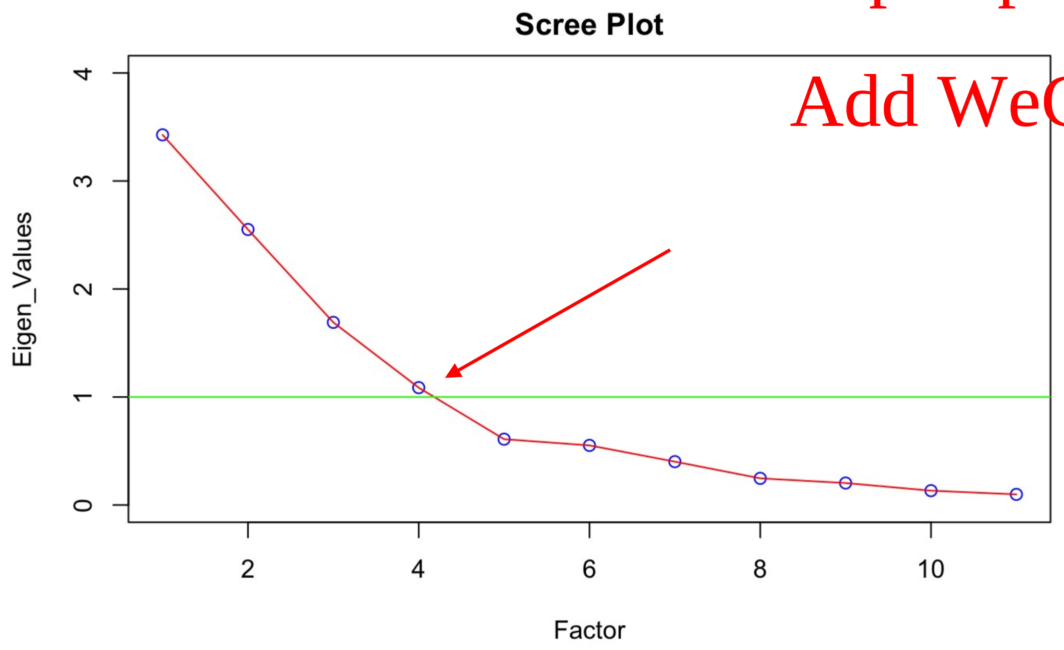
Factor Analysis in R

Step 4 Number of Factors

- Calculate eigen values
- Plot eigen values in a scree plot
- Determine Number of factors


```
ev <- eigen(cor(data_fa))
ev$values
```

```
[1] 3.42697133 2.55089671 1.69097648 1.08655606 0.60942409 0.55188378
[7] 0.40151815 0.24695154 0.20355327 0.13284158 0.09842702
```

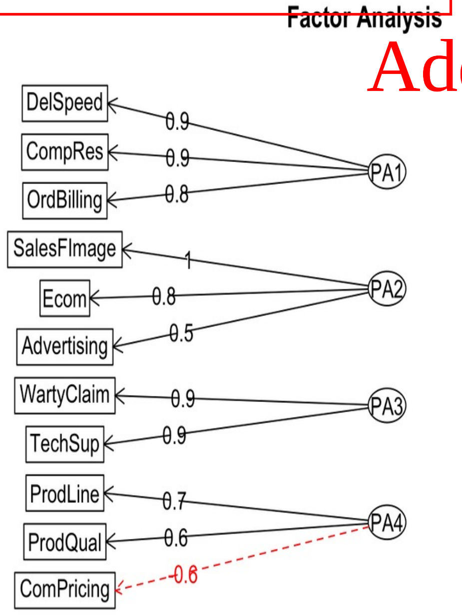



Factor Analysis in R

Step 5 Run Factor Analysis

```
nfactors <- 4
fit1 <-factanal(data_fa,nfactors,scores =
c("regression"),rotation = "varimax")
print(fit1)

fa_var <-  fa(r=data_fa, nfactors = 4,
rotate=”varimax",fm="pa")
fa.diagram(fanone)
```



Loadings:

	Factor1	Factor2	Factor3	Factor4
ProdQual				0.557
Ecom		0.793		
TechSup			0.872	0.102
CompRes	0.884	0.142		0.135
Advertising	0.190	0.521		-0.110
ProdLine	0.502		0.104	0.856
SalesFImage	0.119	0.974		-0.130
CompPricing		0.225	-0.216	-0.514
WartyClaim			0.894	0.158
OrdBilling	0.794	0.101	0.105	
DelSpeed	0.928	0.189		0.164

	Factor1	Factor2	Factor3	Factor4
SS loadings	2.592	1.977	1.638	1.423
Proportion Var	0.236	0.180	0.149	0.129
Cumulative Var	0.236	0.415	0.564	0.694

Factor Analysis in R

Step 6 Regression

- Extract scores from factor analysis
- Combine response and predictors
- Label factors

```
head(fa_var$scores)
```

```
            PA1        PA2          PA3         PA4
[1,] -0.1338871  0.9175166 -1.719604873  0.09135411
[2,]  1.6297604 -2.0090053 -0.596361722  0.65808192
[3,]  0.3637658  0.8361736  0.002979966  1.37548765
[4,] -1.2225230 -0.5491336  1.245473305 -0.64421384
[5,] -0.4854209 -0.4276223 -0.026980304  0.47360747
[6,] -0.5950924 -1.3035333 -1.183019401 -0.95913571
```


Factors	Variables	Label
PA1	DelSpeed, CompRes, OrdBilling	Purchase
PA2	SalesFImage, Ecom, Advertising	Marketing
PA3	WartyClaim, TechSup	Post Purchase
PA4	ProdLine, ProdQual, CompPricing	Product Position

```
regdata <- cbind(data_X[12], fa_var$scores)
#Labeling the data

names(regdata) <- c("Satisfaction", "Purchase", "Marketing",
                    "Post_purchase", "Prod_positioning")
```

	Satisfaction <dbl>	Purchase <dbl>	Marketing <dbl>	Post_purchase <dbl>	Prod_positioning <dbl>
1	8.2	-0.1338871	0.9175166	-1.719604873	0.09135411
2	5.7	1.6297604	-2.0090053	-0.596361722	0.65808192
3	8.9	0.3637658	0.8361736	0.002979966	1.37548765
4	4.8	-1.2225230	-0.5491336	1.245473305	-0.64421384
5	7.1	-0.4854209	-0.4276223	-0.026980304	0.47360747
6	4.7	-0.5950924	-1.3035333	-1.183019401	-0.95913571

Factor Analysis in R

Step 6 Regression (cont)

- Split data in train 0.7 and test 0.3
- Train model

```
set.seed(100)
indices= sample(1:nrow(regdata), 0.7* nrow(regdata))
train=regdata[indices,]
test = regdata[-indices,]
```



```
#Regression Model using train data
model1 = lm(Satisfaction~., train)
summary(model1))
```

```
Call:
lm(formula = Satisfaction ~ ., data = train)

Residuals:
    Min      1Q  Median      3Q     Max 
-1.6857 -0.4018  0.1051  0.4027  1.2036 

Coefficients:
                Estimate Std. Error t value Pr(>|t|)    
(Intercept)      6.92625    0.08263  83.827  < 2e-16 ***
Purchase         0.62022    0.08408   7.377 3.73e-10 ***
Marketing        0.57735    0.08047   7.175 8.50e-10 ***
Post_purchase    0.09567    0.08667   1.104    0.274    
Prod_positioning 0.66562    0.09374   7.101 1.15e-09 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.6814 on 65 degrees of freedom
Multiple R-squared:  0.7079,	Adjusted R-squared:   0.69 
F-statistic: 39.39 on 4 and 65 DF,  p-value: < 2.2e-16
```

Checking for multicollinearity VIF

```
vif(model1)
```

```
Purchase         Marketing    Post_purchase Prod_positioning 
1.012217          1.009683         1.009037         1.012533 
```

Factor Analysis in R

Step 7 Prediction

```
library(Metrics)
pred_test1 <- predict(model1, newdata = test, type = "response")

test$Satisfaction_Predicted <- pred_test1
head(test[c(1,6)], 10)
```


	Satisfaction <dbl>	Satisfaction_Predicted <dbl>
1	8.2	7.269232
2	5.7	7.158146
3	8.9	8.550469
4	4.8	5.541333
5	7.1	6.690958
7	5.7	4.661277
14	7.6	7.963941
21	5.4	5.570249
23	7.0	7.704405
27	6.3	7.361437