



Prediktiv analys

FÖRELÄSNING 4

Dagens fråga

VILKEN SVENSK STAD BORDE
INTE INKLUDERAS PÅ KARTAN?
VARFÖR?



Dagens agenda

- ♦ Regression
- ♦ The multiple regression model
- ♦ Ordinary least square
- ♦ Error metrics för regression
 - Mean squared error
 - Root mean squared error
 - Mean absolute error
 - R-squared
 - Explained variance
- ♦ Predicting crime



Förra föreläsning

- ♦ Viktiga python bibliotek för prediktiv analys:
 - Numpy
 - Pandas
 - Matplotlib
 - Seaborn
 - Scikit-learn
- ♦ Prediktiv analys i python: viktigaste stegen
 1. Förbereda data
 2. Importera estimeringsobjektet
 3. Skapa instans av modellen
 4. Träna modellen
 5. Utvärdera modellen
 6. Prediktera



Vi följer specifika steg för att bygga en Machine learning modell



Detta är generella steg och man kan få hoppa fram och tillbaka flera gånger innan det blir rätt.

Multiple Linear Regression Model



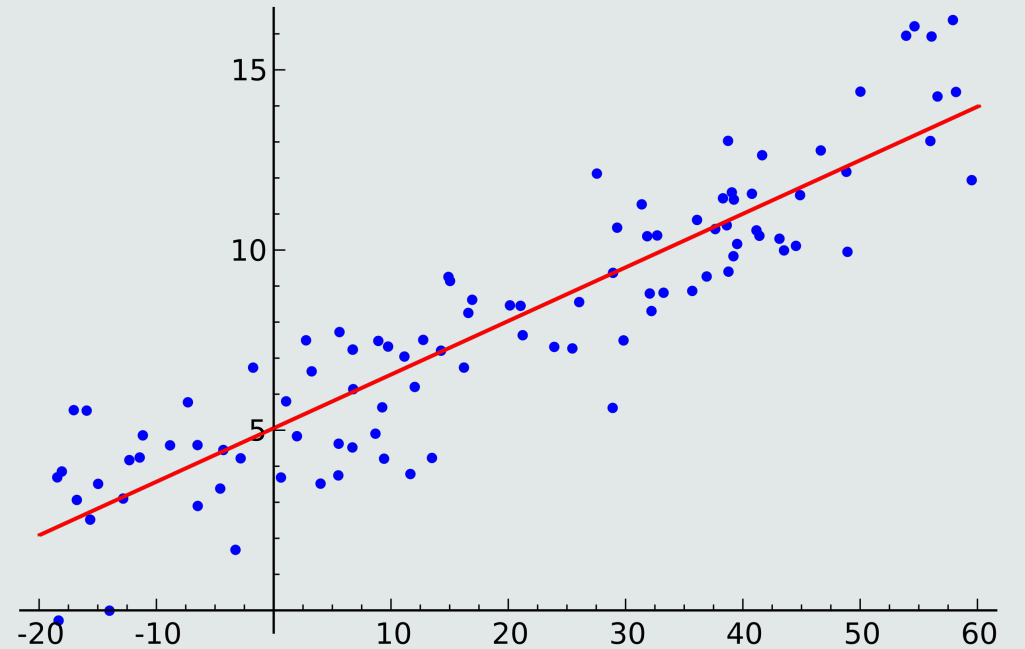
DEN GENERELLA
FORMULERINGEN



KOEFFICIENT
TOLKNING



LINEAR
REGRESSION
ESTIMATOR IN
SCIKIT-LEARN



Den generella formuleringen

- I den här modellen försöker vi prediktera en output y_{pred} (target) med en linjärkombination av p egenskaper x (features)

$$y_{pred} = w_0 + w_1x_1 + w_2x_2 + \dots + w_px_p + \epsilon$$

- Koefficienterna w står för weights
- ϵ är det slumpmässiga felet i modellen mellan de oberoende variablerna x och den beroende variabeln y , alltså modellfelet
- Linear regression beskriver relation mellan variabler mellan att anpassa en linje till den observerade datan
- Man kan estimerar hur den beroende (target) värdet ändrar sig efter som de oberoende (features) ändras
- **Multiple** linear regression när det är två eller fler oberoende features

Kom ihåg inlärningsmodellen

$$\text{Learning Model} = \text{Model} + \text{Algorithm}$$

Modell: Den generella formuleringen av förhållandet mellan "features" och "target"

Learning Algorithm: Tillvägagångssättet för att finna den **specifika formen** för *Modellen*, vanligtvis genom inläring av parametrar från data

Model	Learning Algorithm
Linear Regression	Ordinary Least Squares Method
Logistic Regression	Gradient descent
Artificial Neural Networks	Back propagation
Support Vector Machines	Quadratic programming
Perceptron	Perceptron learning algorithm

Ordinary least square

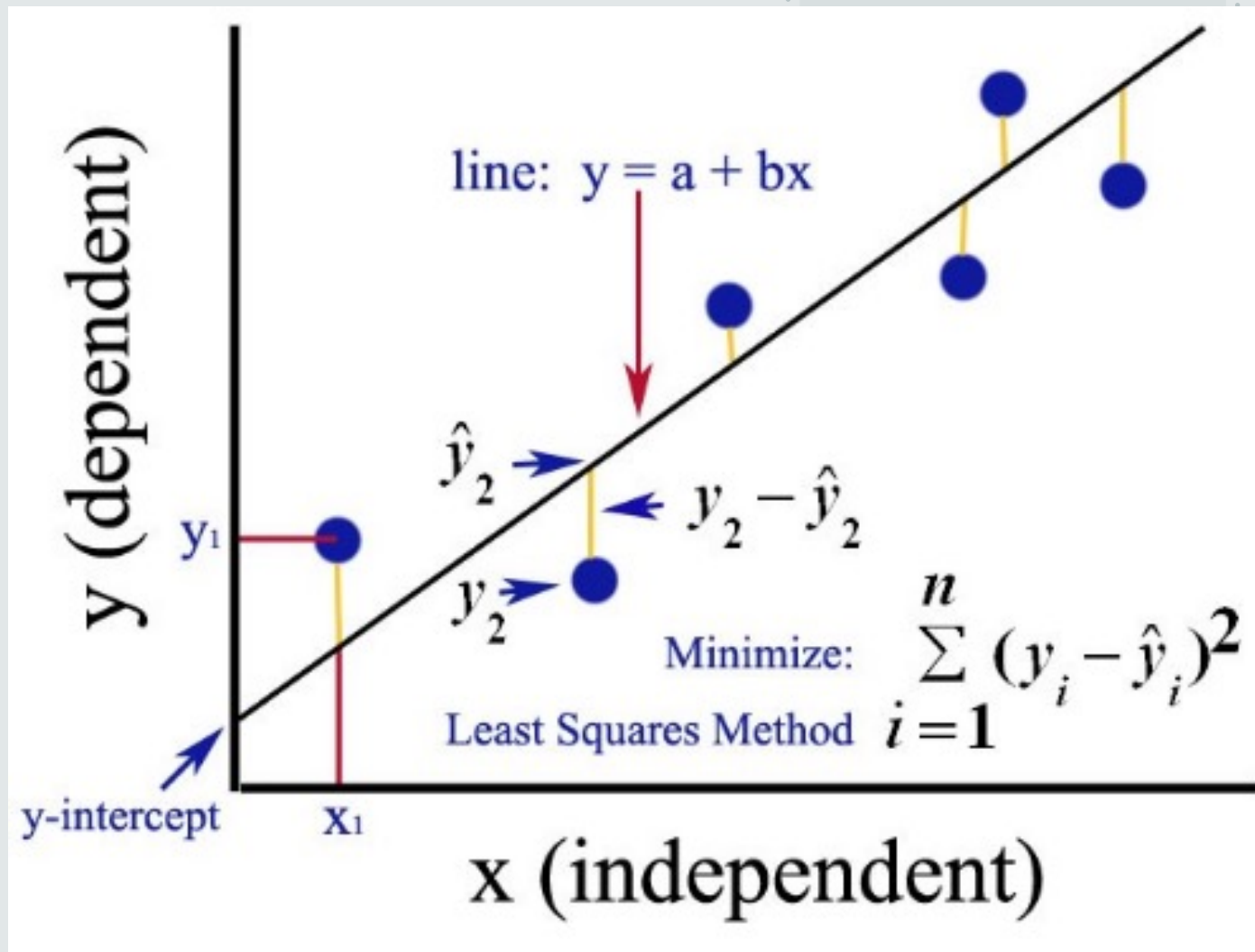
$$y_{pred} = w_0 + w_1x_1 + w_2x_2 + \dots + w_px_p + \epsilon$$

- Denna modellen används för att producera "Ordinary Least Squares" för att hitta "w" sådan att följande formels kvantitet blir minimal
- Ordinary least squares metoden minimerar summan av kvadraten av skillnaderna mellan observerade och predikterade värden

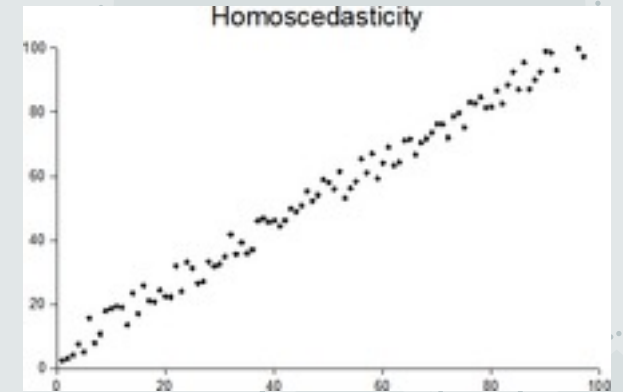
$$RSS = \sum_{i=1}^n (y_{pred_i} - y_i)^2$$

Ordinary least square

- Minimera avståndet mellan predikterat y_{pred} och sanna y
- Det blir ett optimeringsproblem att hitta bäst vikter w för att y_{pred} och sanna y ska vara så när som möjligt



Antaganden



Multiple linear regression gör en del antaganden om datan:

- **Homogen varians** – storleken på felet i prediktionen relativt lika över värdena för de oberoende variablerna. *Konstant fel*
- **Oberoende observationer** – variablerna är inte beroende av varandra *Ingen korrelation mellan variablerna, viktigt att kolla innan!*
- **Normalitet** – felet till modellen ϵ är normalfördelad. Gäller för små dataset (<200 punkter) för eller gäller central limit theorem som säkerställer att felet är normalfördelat
- **Linjäritet** – det måste finnas ett linjärt samband mellan output variabel och de oberoende input variablerna. *Scatterplot för att se linjärt samband mellan output och input variabler*

Exempel från Kriminalitet exempel

$$y_{pred} = w_0 + w_1x_1 + w_2x_2 + \dots + w_px_p$$

- Från crime datan blir detta:

$$\begin{aligned} y_{pred} \\ = w_0 + w_1householdsize + w_2pctUrban + w_3medIncome + w_4PctKids2Par \\ + w_5PctIlleg \end{aligned}$$

- Efter att vi hittade koefficienterna i steg 05 så ser modellen ut ut så här:

$$\begin{aligned} y_{pred} \\ = 0.47 + 0.11householdsize + 0.05pctUrban + 0.09medIncome - 0.67PctKids2Par \\ + 0.27PctIlleg \end{aligned}$$


Koefficient tolkning

- Medelförändringen i målet (target) när respektive egenskap (feature) förändras i en enhet.
- Förutsatt att alla andra faktorer är samma
- Tex "PctKids2Par": -0.67
 - Andel barn i familjer med två föräldrar
 - Om värdet ökar (mer positivt värde) så går kriminaliteten ner
 - Om värdet minskar (mer negativt värde) så går kriminaliteten upp

Linear Regression estimator in scikit-learn

Koefficienterna (weights)
kan fås ut med
funktionsattributet "coef_"

Den första koefficienten
(kallad "intercept" w_0) kan
fås ut efter träning av
modellen med
funktionsattributet
"intercept_"



Saker att tänka på med ”Linear Regression Model”

- Trots ordet ”Linear” så kan man använda modellen när relationen mellan egenskaper (features) och målet (target) är olinjärt (genom att tex addera icke linjära termer)
- Det är enkelt att förklara och förstå
- Behöver inte mycket data för att tränas
- Påverkas av värden i periferin (för-behandling/-beredning av data är viktigt)
- Är inte så flexibel och har inte så mycket kraft till att prediktera komplexa problem

Modell utvärdering – error metrics

- Det finns många parametrar som kan användas för att evaluera prestandan på modellen
- Hur nära är alltså vårt predikterade target \hat{y} jämfört med det faktiska sanne värdet på y
- Målet är att mäta hur nära de predikterade värdena hamnar gent emot de observerade målvärdena
- Alla dessa finns i sklearn.metrics modulen

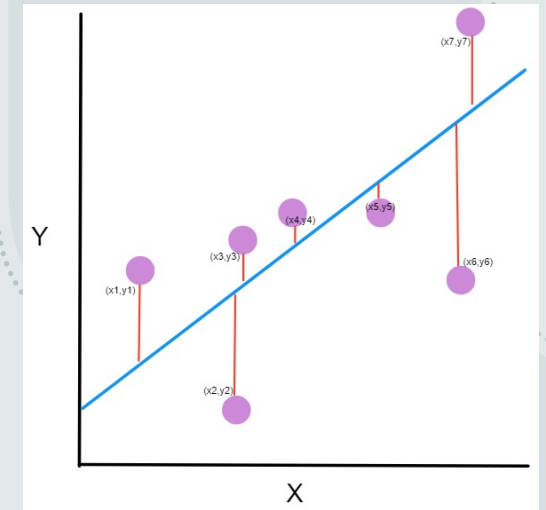
Error metric	Scikit-learn
MSE	mean_squared_error()
RMSE	mean_squared_error(squared=False)
MAE	mean_absolute_error()
R^2	r2_score()
Explained variance	explained_variance_score()

Mean Squared Error

- Mean squared error (MSE):

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

- Där y_i är observerad target värde, \hat{y}_i är motsvarande predikterat target, n är antal observationer
- Genomsnittsfelen till alla observationer
- Avståndet i kvadrat så att tecken inte spelar någon roll. $(-1)^2 = 1$ och $1^2 = 1$
- Blir summan av alla fel för alla mätpunkter för hela modellen så kan bli ett väldigt högt tal!
- Mäter medelvärde för kvadratroten ur fel eller avvikelser
- Ju mindre skillnaden mellan det observerade och predikterade värdet är, desto bättre blir modellen, så lågt MSE är bra

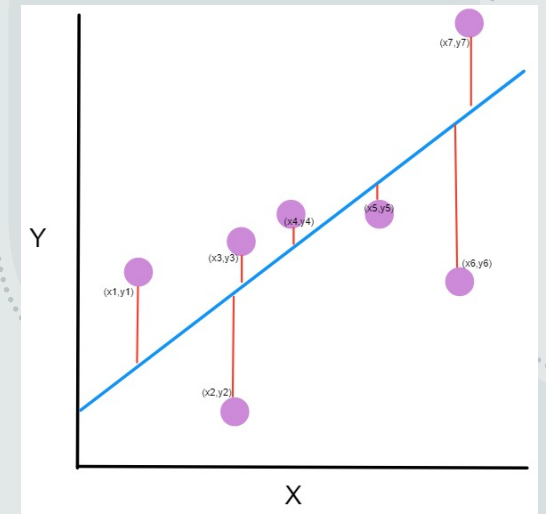


Root Mean Squared Error

- Root Mean Squared Error (RMSE)

$$RMSE = \sqrt{MSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2}$$

- Där y_i är observerad target värde, \hat{y}_i är motsvarande predikterat target, n är antal observationer
- Kvadratroten ur Mean squared error
- Mer tydligt resultat eftersom det har samma skala som observationerna
- Lågt värde är bra

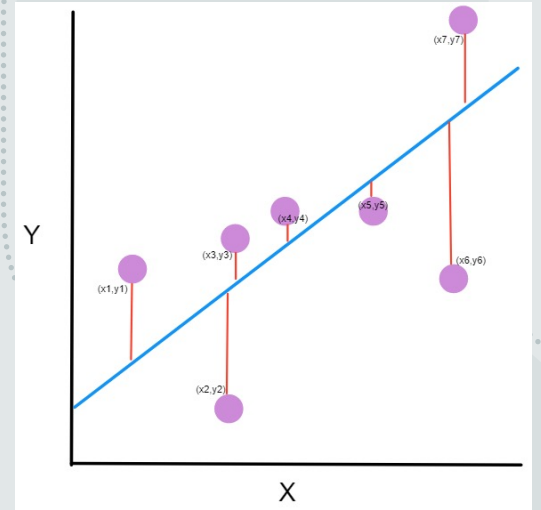


Mean Absolute Error

- Mean absolute error (MAE)

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|$$

- Där y_i är observerad target värde, \hat{y}_i är motsvarande predikterat target, n är antal observationer
- Mäter genomsnitt av storleken på felet
- Samma skala som observationerna
- Lågt värde är bra



Skillnad RMSE och MAE

- Att ta kvadratroten av de genomsnittliga kvadratiska felen har några intressanta konsekvenser för RMSE
- Eftersom felen kvadreras innan medelvärdet beräknas ger RMSE en relativt hög vikt åt stora fel
- Detta innebär att RMSE borde vara mer användbar när stora fel är särskilt oönskade
- Tabellerna visar att MAE är konstant och RMSE ökar när det finns större varians i distributionen av felen
- Beroende på hur de beräknas kommer alltid

$$MAE \leq RMSE$$

CASE 1: Evenly distributed errors				CASE 2: Small variance in errors				CASE 3: Large error outlier			
ID	Error	Error	Error^2	ID	Error	Error	Error^2	ID	Error	Error	Error^2
1	2	2	4	1	1	1	1	1	0	0	0
2	2	2	4	2	1	1	1	2	0	0	0
3	2	2	4	3	1	1	1	3	0	0	0
4	2	2	4	4	1	1	1	4	0	0	0
5	2	2	4	5	1	1	1	5	0	0	0
6	2	2	4	6	3	3	9	6	0	0	0
7	2	2	4	7	3	3	9	7	0	0	0
8	2	2	4	8	3	3	9	8	0	0	0
9	2	2	4	9	3	3	9	9	0	0	0
10	2	2	4	10	3	3	9	10	20	20	400
MAE 2.000 RMSE 2.000				MAE 2.000 RMSE 2.236				MAE 2.000 RMSE 6.325			

R-squared

- R-squared, R^2

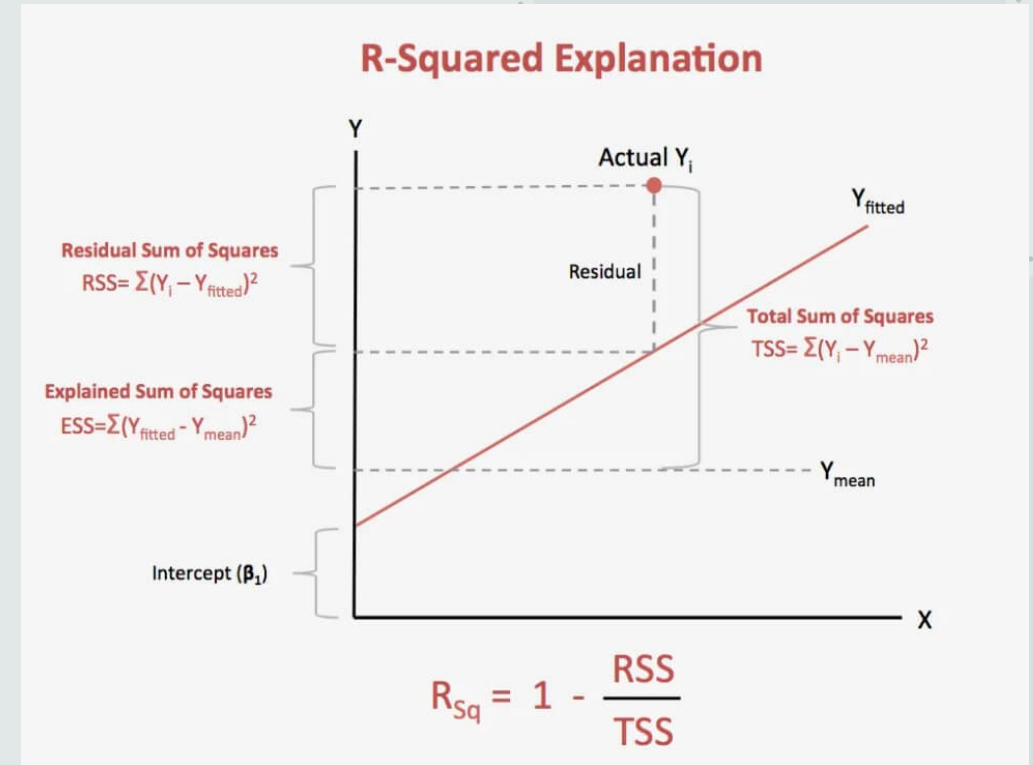
$$R^2 = 1 - \frac{RSS}{TSS}$$

Var

$$RSS = \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$TSS = \sum_{i=1}^n (y_i - \bar{y})^2$$

- Och y_i är observerad target värde, \hat{y}_i är predikterat värde, \bar{y} är genomisnittet av y_i och n är antal observationer
- RSS också kallad det oförklarliga felet är variation i y som inte fångas upp av modellen
- R^2 är coefficient of determination som betyder att den mäter hur mycket av variansen som förklaras av modellen
- R^2 är tal mellan 0 och 1 och vi vill vara så nära 1 som möjligt



Adjusted R-squared

$$R_{adj}^2 = 1 - \frac{(1 - R^2)(n - 1)}{n - p - 1}$$

- Där n är antal datapunkter och p är antal features i modellen
- För att R-squared ökar eller har samma värde när antal features i modellen ökar så kan detta vara missvisande
- Adjusted R-squared ökar alltså (vilket vi önskar) om nya features förbättrar modellens förmåga att prediktera

Explained variance

$$\text{explained variance} = 1 - \frac{\text{Var}(y - \hat{y})}{\text{Var}(y)}$$

- Där $\text{Var}(y - \hat{y})$ är variansen är prediktionsfelen och $\text{Var}(y)$ är variansen till target y
- Berättar hur mycket modellen står för variansen (spridningen) hos datan
- Värde mellan 0 och 1 och vi vill vara nära 1

Länkar

- ♦ Multiple linear regression <https://statsandr.com/blog/multiple-linear-regression-made-simple/>
- ♦ Ordinary least square <https://towardsdatascience.com/understanding-the-ols-method-for-simple-linear-regression-e0a4e8f692cc>
- ♦ Error metrics <https://machinelearningmastery.com/regression-metrics-for-machine-learning/>
- ♦ Skillnad MAE och RMSE <https://medium.com/human-in-a-machine-world/mae-and-rmse-which-metric-is-better-e60ac3bde13d>

Vad har vi gjort idag?

- ♦ Regression
- ♦ The multiple regression model
- ♦ Ordinary least square
- ♦ Error metrics för regression
 - Mean squared error
 - Root mean squared error
 - Mean absolute error
 - R-squared
 - Explained variance
- ♦ Predicting crime



Nästa lektion

- ♦ Regression i Python
- ♦ KNN
- ♦ Model evaluation för regression
 - Cross validation
 - Overfitting
 - Bias-variance tradeoff
 - Overfitting
 - Regularization
 - Feature selection
- ♦ Lasso regression
- ♦ Diamond prices
- ♦ Post popularity

