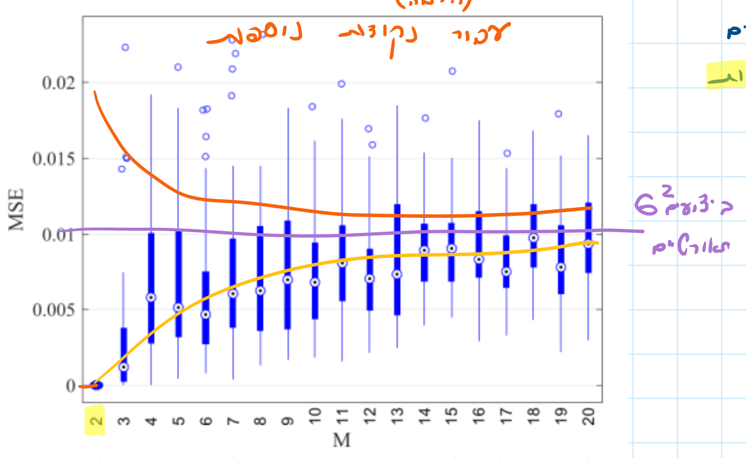
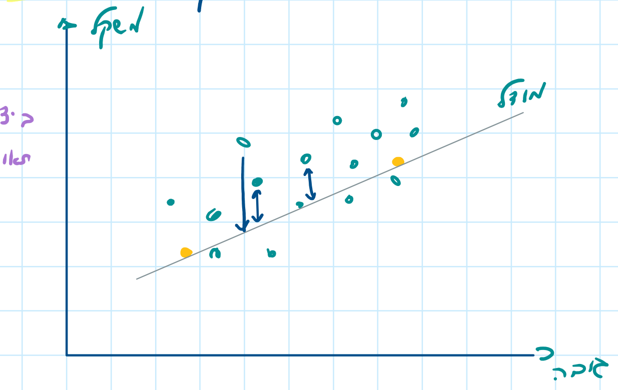


למה יותר נקודות ⇐ ביצועים פחות?

בתרגיל בית: התעלמות בבדיקת ביצועים

מיתר הנקודות אפשריות



מטרות:

* מגבלות של בדיקת ביצועים
* ניתוח ביצועים
* ביצועים של מעשה "הנדשם"

Generalization (הכללה)

בדיקת ביצועים: תאוריה ← מעשית כמות נתונים מוגבלת (עד מאוד)

דוגמה חד-ממדית (x_k, y_k)

← נדרוש כל אפשרויות הפוטנציאליות של (x_k, y_k) עם הסתברות שלהם

⇐ ביצועים: ממוצע עבור כל זוג x_k, y_k כפול הסתברות

$$\underline{\text{ממוצע}} = \left[MSE(x_k, y_k) \cdot \text{הסתברות של הזוג } x_k, y_k \right]$$

← מודל

שאלה: איך יודעים עד כמה קרובים לביצועים תאורטיים?

הערה: תוצאה מדוייקת עבור מספר מצומצם של דוגמאות סינטטי.

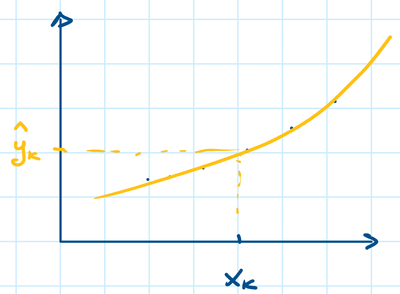
Polynomial model

Goal: • Extension of a linear model "engine" to polynomial models. The polynomial model is very flexible, i.e. due to the Taylor expansion theorem.

• Illustration of generalization principle.

The N-degree uni-variate polynomial model is

$$\hat{y} = f(x; \mathbf{w}) = w_0 + w_1 x + w_2 x^2 + \cdots + w_N x^N$$
$$= \sum_{j=0}^{N} w_j x^j$$



M נקודות

$k = 1, \ldots, M$

$$\hat{y}_k = \sum_{j=0}^{N} w_j x_k^j$$

N - הזקה מקסימלית של פולינום

לימוד המודל:

* להפוך מודל חד-ממדי פולינומיאלי למודל ליניארי רב-ממדי.

⇐ w_j

$$\begin{array}{c} \quad x_1 \quad x_2 \quad \cdots \quad x_N \\ \mathbf{X} = \begin{bmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^N \\ 1 & x_2 & x_2^2 & \cdots & x_2^N \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_M & x_M^2 & \cdots & x_M^N \end{bmatrix} \end{array}$$

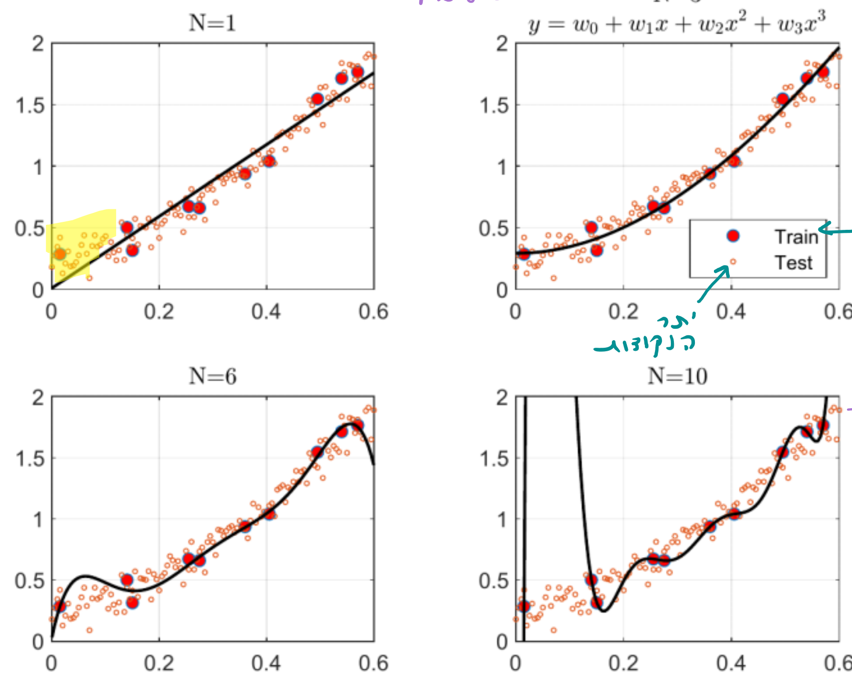
$$\hat{y}_k = \sum_{j=0}^{N} w_j z_{kj} \qquad z_{kj} = x_k^j$$

$\begin{bmatrix} 1 & x_M & x_M^2 & \cdots & x_M^N \end{bmatrix}$ $\quad j=0$

רב-ממדי

$w_j \Leftarrow$

N סדר של פולינום



שים לב
לחישוב
מקדמי
הפולינום

יותר
נקודות

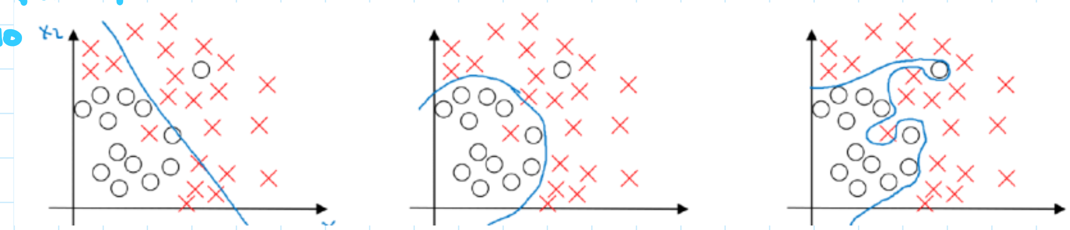
עובר על
כל הנקודות

למידה מקדמים

hyper-parameters
= פרמטרים של
למידה שיכול
לחשב המודל
דוגמא: סדר של
N פולינום
N מקדמים של
פולינום.פרמטר

קו המפריד בין צבע
סוגים של מידע

דוגמה: ביטויי סיווג



3.3 Overfitting and underfitting

Goal: Two common and fundamental אתגרים in machine learning.

Overfitting when model is too complex, i.e. have too many parameters.

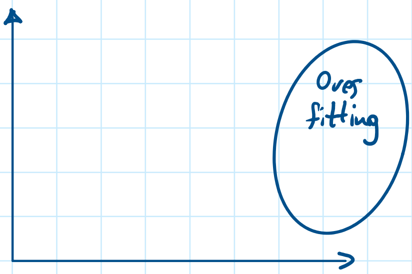
N=10

- Too many hyper-parameters relative to the number of observations. יותר מדי פרמטרים
- Follow the training data very closely. עוקב יותר מדי טוב
- Fail to generalize well to unseen data.

Underfitting happens when a model is too simple.

- Unable to capture the underlying pattern of the data and hence misses the trends in the data. לא תופס מורכבות של המודל
- Performs poorly on the training data and fail to generalize. ביצועים גרועים

Overfitting and underfitting are complimentary and balancing between them is key to building robust machine learning models that perform well on new, unseen data, i.e. generalize well.



Hyper-parameter optimization The order N is the hyper-parameter of the polynomial model. Selecting the most appropriate hyper-parameters value is called hyper-parameter optimization.

בדיקת ביצועים

Cross-validation

בדיקת ביצועים

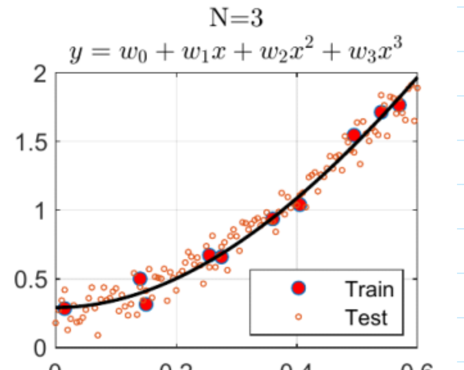
Cross-validation

ניסוי וטעייה

Goal: Trial and error approach to quantify generalization performance and overfitting-underfitting balance.

The cross-validation is also termed performance assessment.

- First step of any following technique is resample the dataset into the random order.



train test

מצב לא תקין

10k+

Big dataset (tens of thousands): train/validation/test

חלקים נפרדים

Split into three *distinctive* datasets:

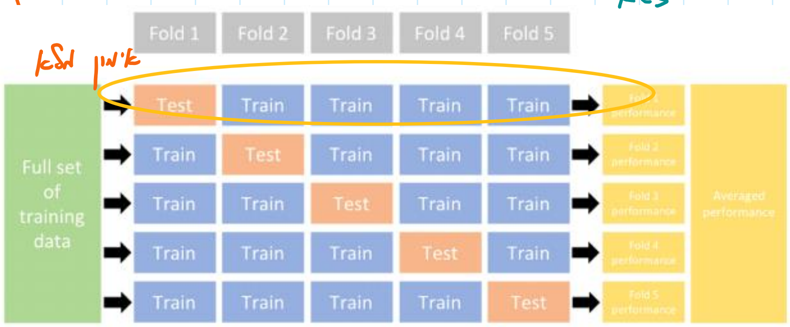
1) - *Training* (50-80%): used for learning of model parameters, e.g. weights **w**.
2) - *Validation* (10-25%): used for assessment of model hyper-parameters influence.
3) - *Test* (10-25%): performance assessment that is supposed to be an estimation for the generalization error.

Medium datasets (100 - 1k)

k-fold

k=5

אימון מלא



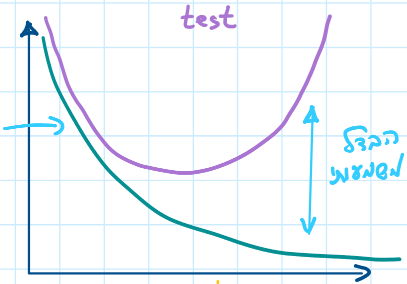
1) חלוקה ל-k חלקים שווים

2) כל פעם, חלק אחד הוא test

בד״כ k=5,10

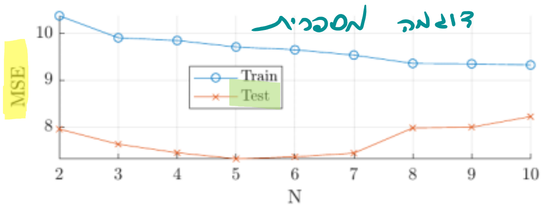
עבור בסיס נתונים קטנים במיוחד k=M

one-hold-out



מודל פולינומיאלי

דוגמה מספרית



test: מטרה היא בדיקת המודל

Bias-variance trade-off

הנחת יסוד עבור הנתונים

פונ׳ לא ידועה

$$y = h(\mathbf{x}) + \epsilon$$

נתונים רעש

* רעש עם ממוצע 0 ועם שונות σ^2

פונ׳ מחיר: MSE

$$\mathcal{L} = E\left[(\hat{y}-y)^2\right] = \frac{1}{M}\sum_{i=1}^{M}(\hat{y}_i - y_i)^2$$

$$= \underbrace{\left(E[\hat{y}] - h(\mathbf{x})\right)^2}_{bias^2} + \underbrace{\mathrm{Var}[\hat{y}]}_{variance} + \underbrace{\sigma^2}_{noise}$$

סכ״ה ממוצע של החיזוי

שונות של החיזוי

רעש

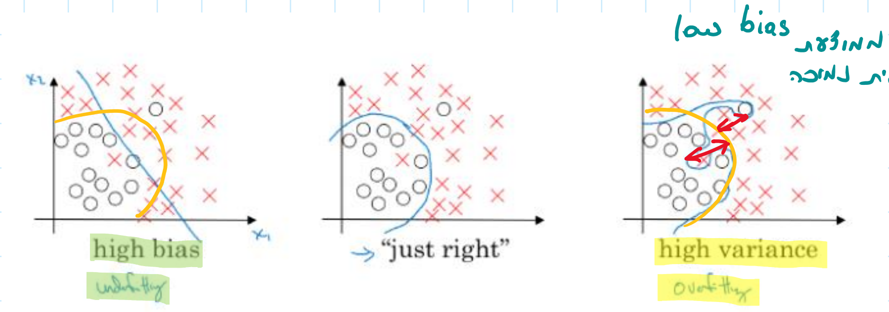
* רעש עם ממוצע 0
ועם שונות σ^2

* מטרה של המודל: ללמוד $h(x)$

סטיה ממוצעת של החזוי — שונות של החזוי

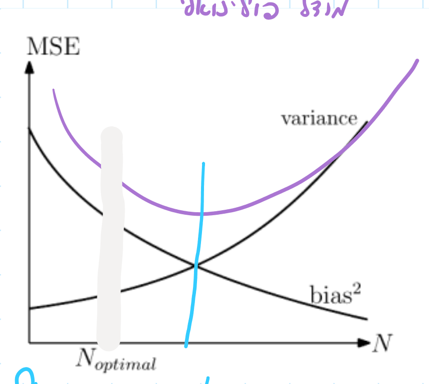
תמורה trade-off

סטיה ממוצעת גבוהה



low bias
* סטיה ממוצעת יחסית נמוכה
* שונות גבוהה

מודל פולינומיאלי



מקרה אופטימלי

מודל ה' - דוגמה מספרית:

```
M_train = 10;                                      train
sigma = 0.1;                                       שונות הרעש
x_train = random('Uniform',0,0.6,M_train,1);       x ~ U[0,0.6]  אקראי
p_coeff = [0.1 3 1 0.2 0.01];                      h(x)
y_theory_train = polyval(p_coeff,x_train);
y_train = y_theory_train  + sigma*randn(M_train,1); <- הוספת רעש
%y = x1 + 3*x1.^2 + 0.1*x1.^3 = h(x)
M_test = 100;  test
x_test = random('Uniform',0,0.6,M_test,1);
y_theory_test = polyval(p_coeff,x_test);
y_test = y_theory_test  + 0.1*randn(M_test,1);
```

p_1 p_5

p_coeff = [0.1 3 1 0.2 0.01];

ראשון — אחרון

$$p(x) = p_1 x^{n} + p_2 x^{n-1} + \dots + p_n x + p_{n+1}$$

$$h(x) = 0.01 + 0.2x + x^2 + 3x^3 + 0.1x^4 \rightarrow y = h(x) + \text{רעש}$$

פונקציה לחישוב מקדמי הפולינום

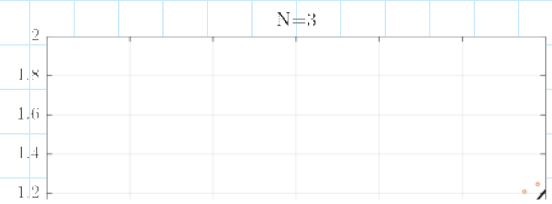
```
function [yh, w] = polynomial_regression_weights(x,y,N)
  M = size(x,1);             <- שורות
  X = zeros(M,N+1);          שורות, עמודות

  for k = 0:N
     X(:,k+1) = x.^k;        } הוספת עמודות
  end
  w = pinv(X)*y; % lsqminnorm(X,y)   <- מקדמי הפולינום
  yh = X*w;                  <- חיזוי ŷ_k עבור x_k
end
```

$N+1$ עמודות

$$M\left\{\begin{array}{ccccc} 1 & x_1 & x_1^2 & \cdots & x_1^N \\ 1 & x_2 & x_2^2 & & \\ 1 & \vdots & & & \end{array}\right.$$

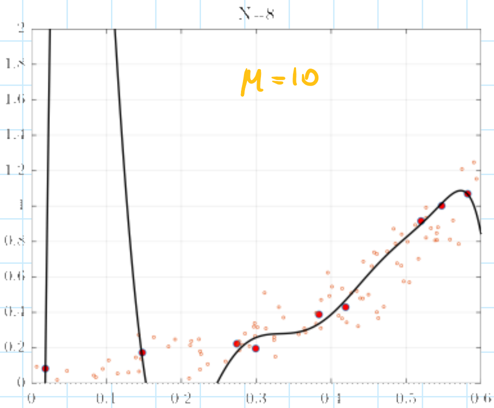
חישוב ערכי $\hat{y}_k$ בהינתן משקלים ו-x_k

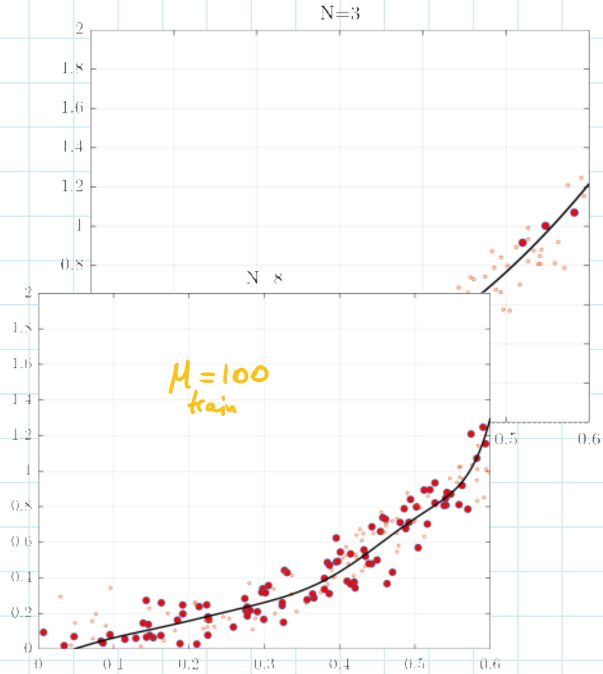
```
function yh = polynomial_regression_values(x,w)
   M = size(x,1);
   N = length(w)-1;
   X = zeros(M,N+1);
   for k = 0:N
      X(:,k+1) = x.^k;
   end
   yh = X*w;                 ŷ = Xw
end
```



```
X = zeros(M,N+1);
for k = 0:N
    X(:,k+1) = x.^k;
end
yh = X*w;
end
```

$\hat{y} = \underline{X}\underline{w}$





* הערה: הוספת נתונים גורמת להפחתת overfit

שיעורי בית

* מקבלים בסיס נתונים.

* יש לחשב/להציג ערך N אופטימלי עבור הנתונים

* שלבים: 1) חישוב מקדמים ע"י train

2) בדיקת ביצועים עבור כל N נבחר ע"י validation ⇐ N אופטימלי

3) סיכום בדיקת ביצועים עבור N_{opt} ע"י test