Review Lecture

Lecture XII

למידת מכונה – סיכום סמסטר

- מקווים שנהנתם ולמדתם... אבל ... כן, יש גם מבחן
  - מבחן– 50% מציון סופי – לגבי הפורמט הודעה בקרוב (או במדול או כל השאלות אמריקאיות כמו במצגת התרגול)
  - תאריך הבחינה 23/6/2021 בשעה 14:00
  - משך המבחן – שעתיים <u>ללא הפסקה</u>
  - <u>המבחן כולל</u> – 25 שאלות, חלקם שאלות הבנה אמריקאיות וחלקם כוללות רכיבי חישוב (או שאלות חישוב)
  - <u>שאלות עם חישוב – כשליש מהשאלות</u>
  - חומר עזר – <u>מחשבון</u> ודף נוסחאות (יינתן בקרוב)
- מיקוד בחומר של הרצאות והתרגיל – <u>לא כולל python</u>
- נושאים מרכזיים:

למידת מכונה – סיכום סמסטר

- נושאים מרכזיים רוחביים:
  - מושגים מתמטיים הסתברותיים וסטטיסטים
  - מידול והבנת בעיות למידת מכונה; flow של למידת מכונה; למידה מונחית מול למידה לא מונחית
  - סילום; טיפול מקדים במידע; אנומליות
  - שיטות מרחק; קשר בין משתנים
  - נושאים מתורת האינפורמציה
  - הערכות ביצועים (סיווג, רגרסיה ו-clustering)
  - אופטימיזציה; Gradient descent
  - בחירת מודל ופרמטרים;

ועוד ...

למידת מכונה – סיכום סמסטר

- Paradigm: “Programming by example”
  - Replace ``human writing code'' with ``human supplying data''
- Most central issue: generalization
  - How to abstract from ``training'' examples to ``test'' examples?

למידת מכונה – מטרות הקורס

- By the end of the semester, you should be able to
  - Look at a problem
  - Identify if ML is an appropriate solution
  - If so, identify what types of algorithms might be applicable
  - Apply those algorithms

- This course is not
  - A survey of ML algorithms
  - A tutorial on ML toolkits such as sklearn, python, …

המאפיינים – ממרחב המאפיינים לווקטור המאפיינים

בעיית האירוסים

- עלי גביע (sepal):
  - אורך, רוחב
- עלי כותרת (petal):
  - אורך, רוחב

sepal		petal	
length	width	length	width
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

בעיית הזברות

Is animal	Vertical stripes	Black & white	4 legs	large
0	1	1	0	0
1	0	1	0	0
1	0	0	1	1
1	1	1	1	1

- חיה: (כן, לא)
- פסים אנכיים: (אנכיים, אופקיים, ללא)
- צבעים: (שחור, לבן, חום, ...)
- רגליים: (2, 4, ללא)
- גודל החיה: (גדולה, בינונית, קטנה)

מאפיינים – ערכים בדידים (מספריים)

מאפיינים – ערכים רציפים (מספריים)

מאפיינים – ערכים קטגוריים.

א. מידול (Modeling):

שאלת סיווג; קטגורית התשובה; **מאפיינים** (features/attribute):

בלמידת מכונה, מתייחסים לחלק מהמאפיינים, כדי למדל דוגמאות (תצפיות).

Feature Set (מרחב המאפיינים): המאפיינים שבאמצעותם נתייחס לכל דוגמה

Feature Vector (וקטור המאפיינים): ערך המאפיינים עבור דוגמה מסוימת (instance)

ערך המאפיין:

- ערך המאפיין יכול להיות קטגורי, מספר בדיד או מספרי רציף
- עבור ערכים קטגוריים, בד"כ (וגם ב-KNN), נרצה להמיר את ערך המאפיין לערך מספרי בדיד.

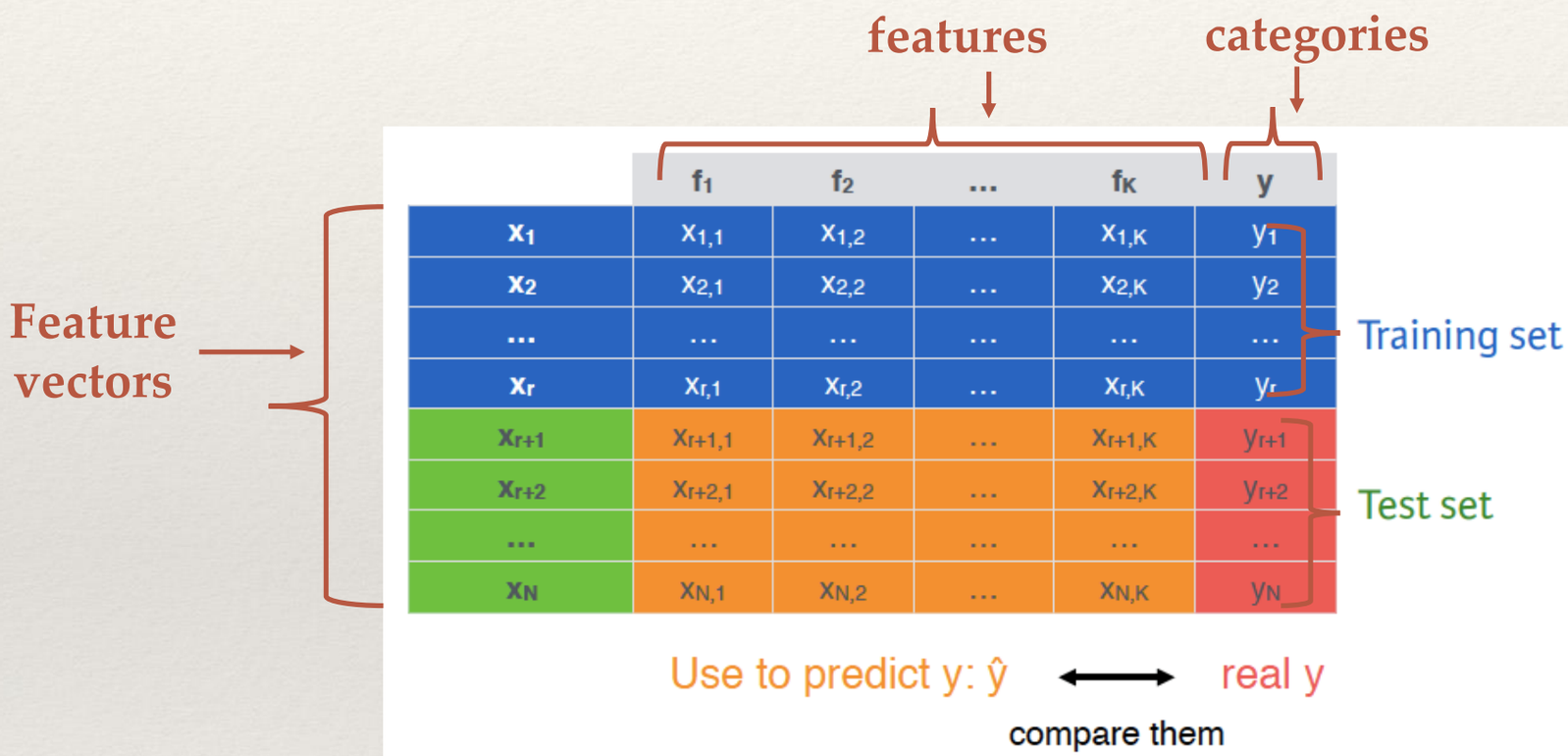
למידת מכונה – מרכיבי למידה

- Training vs. test examples
  - Memorizing the training examples is not enough!
  - Need to generalize to make good predictions on test examples
- Inductive bias
  - Many classifier hypotheses are plausible
  - Need assumptions about the nature of the relation between examples and classes

dataset – train-set and test-set

Training Dataset: The dataset that we use to train the model

- The model *sees* and *learns* from this data.



	f_1	f_2	…	f_K	y
x_1	$x_{1,1}$	$x_{1,2}$	…	$x_{1,K}$	y_1
x_2	$x_{2,1}$	$x_{2,2}$	…	$x_{2,K}$	y_2
…	…	…	…	…	…
x_r	$x_{r,1}$	$x_{r,2}$	…	$x_{r,K}$	y_r
x_{r+1}	$x_{r+1,1}$	$x_{r+1,2}$	…	$x_{r+1,K}$	y_{r+1}
x_{r+2}	$x_{r+2,1}$	$x_{r+2,2}$	…	$x_{r+2,K}$	y_{r+2}
…	…	…	…	…	…
x_N	$x_{N,1}$	$x_{N,2}$	…	$x_{N,K}$	y_N

Test dataset: The dataset that provides the gold standard used to evaluate the model.

- It is only used once a model is completely trained.

שאלות בסיסיות

שאלה 1: למה מתכוונים כשאומרים "שערוך" (evaluation) המודל?

שאלה 2: מהו feature set? ומהו feature vector?

שאלה 3: מהו dataset? מהו train set ומהו test set?

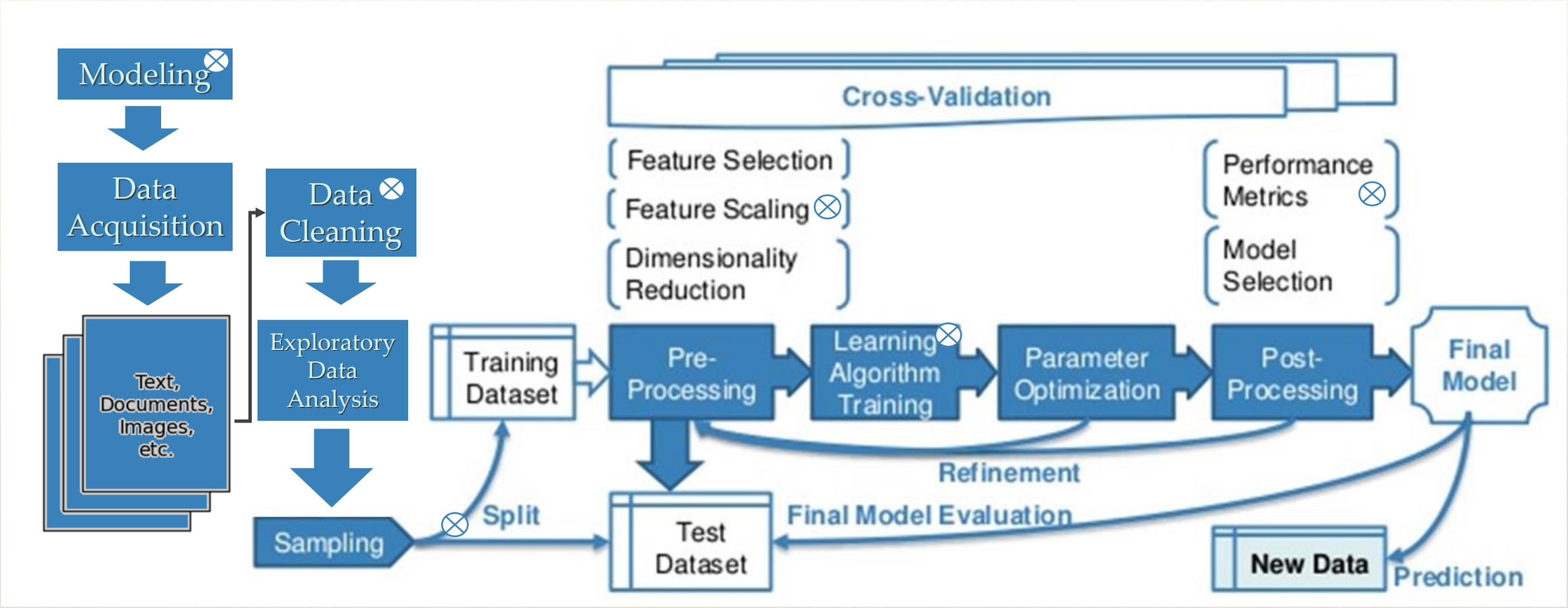
שאלות בסיסיות - המשך

שאלה 4: נניח שאנחנו צריכים לחזות את מזג האוויר מחר (מבחינת טמפרטורה), האם זו בעיית סיווג? מדוע?

שאלה 5: נניח שאנחנו רוצים לבנות מערכת שתחליט אם תמונה שמצלמים במדגסקר (אפריקה) היא תמונה של זברה או שהיא אינה תמונה של זברה. לשם כך אספנו 100 תמונות של זברות בגן החיות ב-'central park' (שבעיר ניו יורק).

- האם התמונות מתאימות (ברמה גבוה) לבניית מודל סיווג? מדוע?

A typical classification flow - diving in



Data Cleaning

→ Duplicates

→ Missing Data

→Remove

→Repair

Train-Test split

+ sampling

Scaling

→ Minmax norm.

→ t-dist. standardization

Feature Selection

→ Feature compared to itself.

→ compared to other features.

→ compared to category.

Learning Algos.

→ KNN

→Decision Trees

→Naïve Bayes

→Perceptron

→ANN

→SVM

Validation

→ Hyper parameter tuning

Post Processing

Evaluation

→ Error (rate)

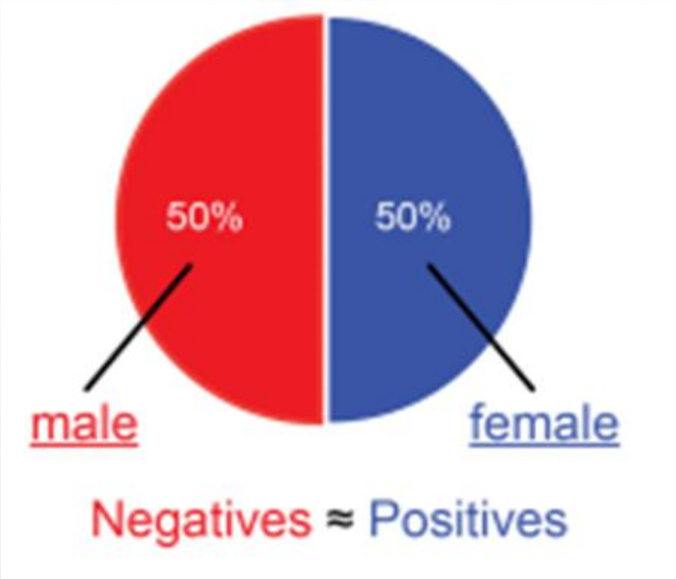
→ Accuracy

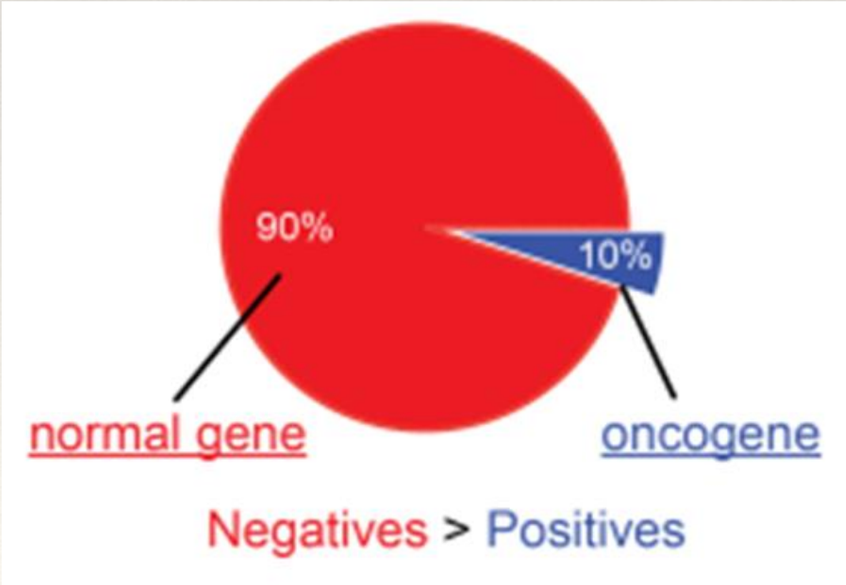
→Confusion matrix

→ Precision

→ Recall

התפלגות המחלקות

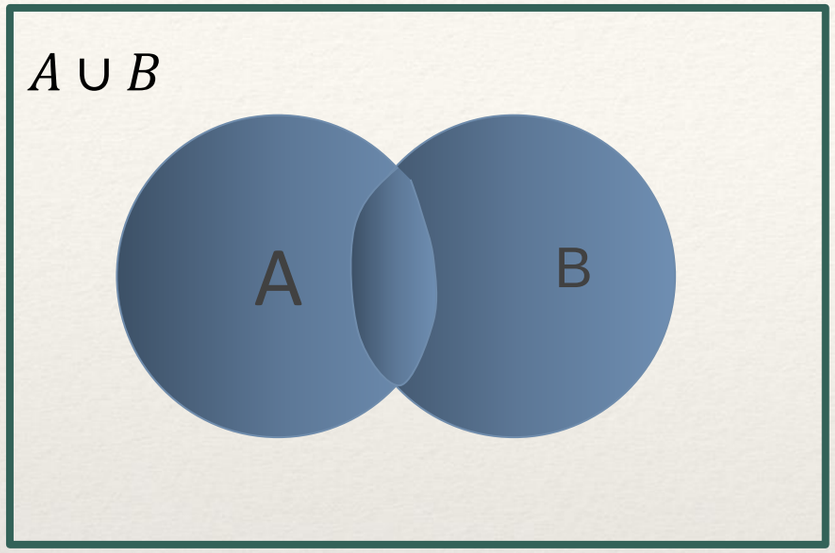


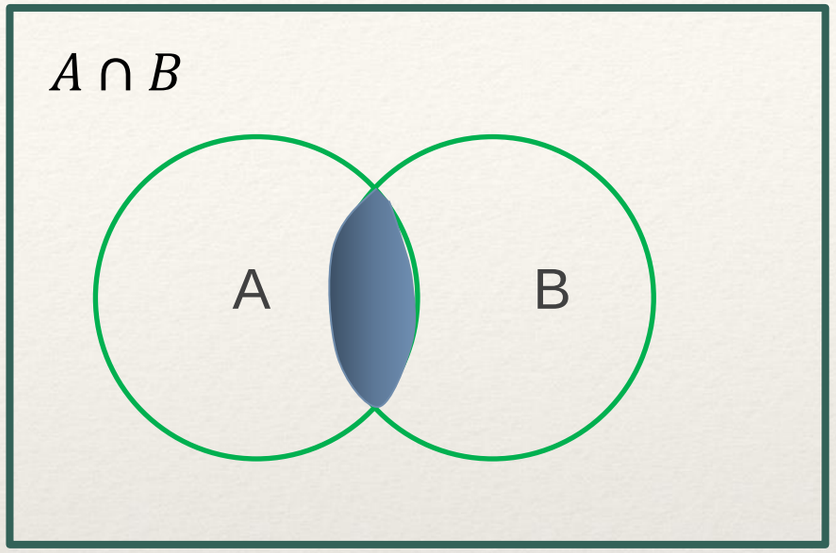


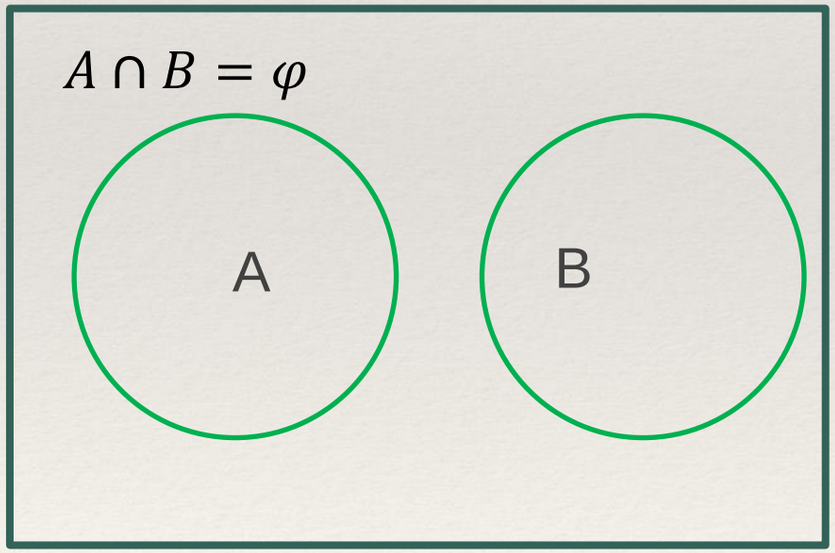
במגדר, למשל, הנתונים מתפלגים בערך בצורה מאוזנת (balanced).

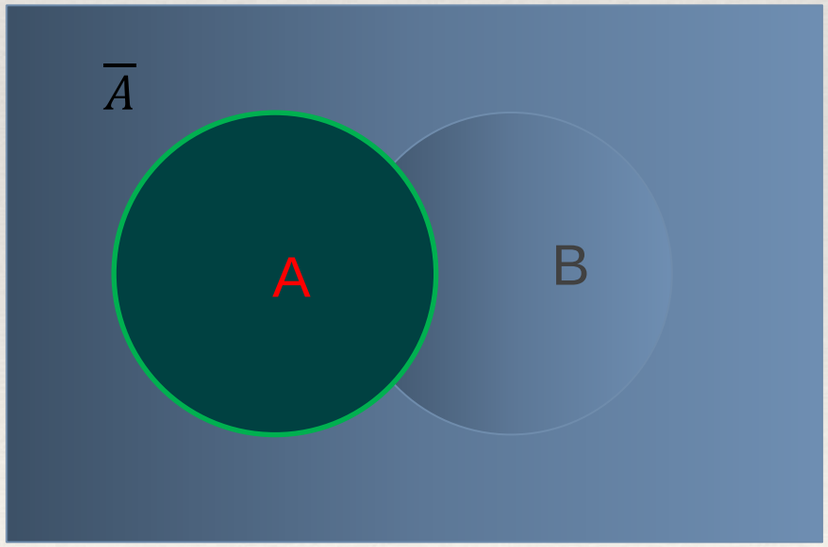
משתנים אחרים, בהם יתכן ונרצה להתייחס (בבעיות סיווג), אינם מתפלגים בצורה מאוזנת (imbalanced)

דיאגרמות ואן









רקע מתמטי וסטטיסטי

סטטיסטיקה - שאלות בסיסיות

שאלה 1: מהם השונות וסטיית התקן? מה הם באים למדוד?

שאלה 2: מה ההבדל בין סטיית תקן באוכלוסיה וסטיית תקן במדגם?

שאלה 3: מהם התפלגות z והתפלגות t, ומה ההבדל בינהם?

סקלרים ווקטורים - שאלות

שאלה 1:

מהו סקלר? מהו וקטור בתצורה גאומטרית? מהו וקטור בתצורה אלגברית?

שאלה 2:

מהי נורמה?

מה המשמעות של מכפלה סקלרית של וקטורים במובן האלגברי?

מה המשמעות של מכפלה סקלרית של וקטורים במובן הגאומטרי?

מהו משמעות כפל וקטורים שנותן 1-?

מהו משמעות כפל וקטורים שנותן 0?

שערוך מודלים - שאלות

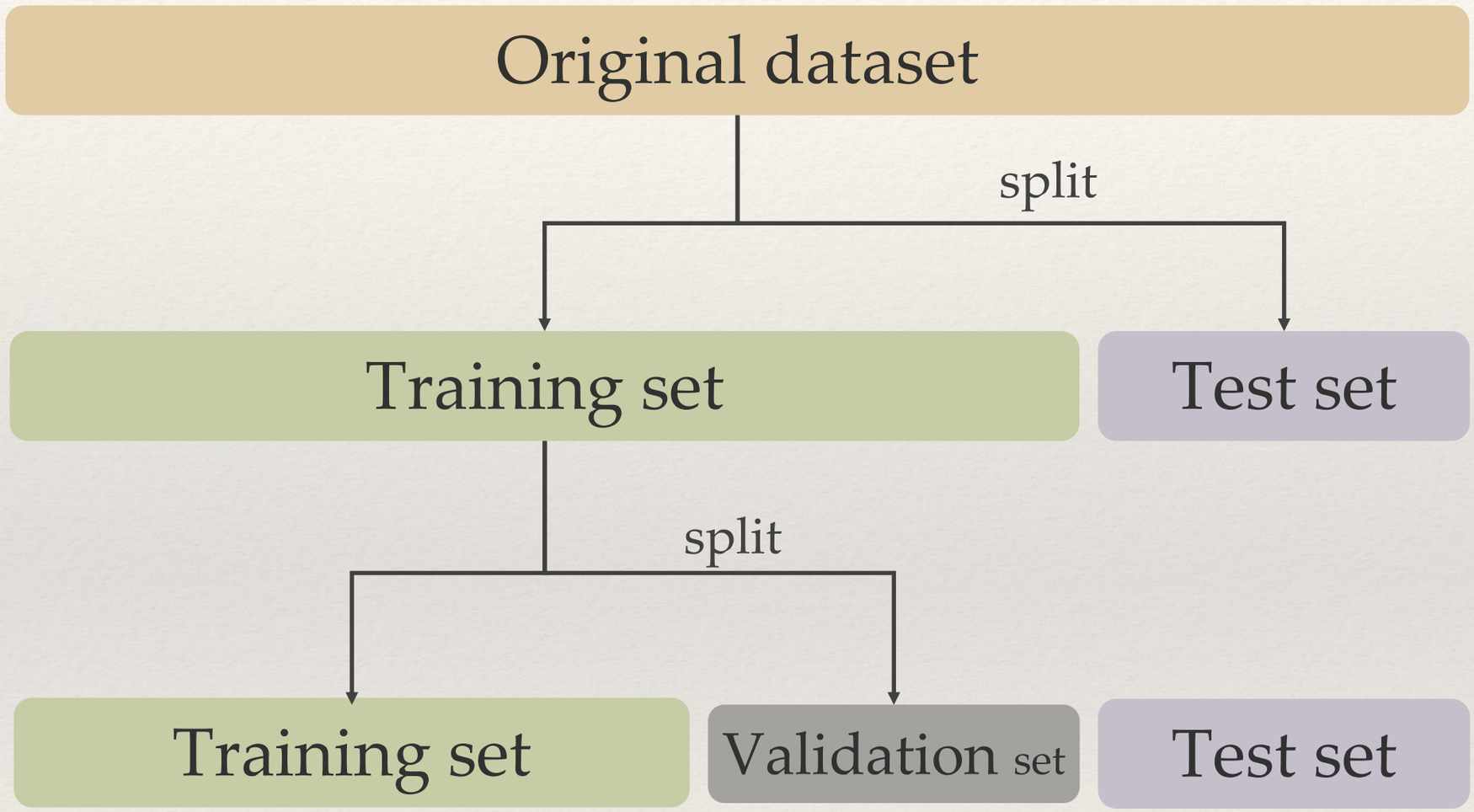
1. אילו שיטות שיערוך מיועדים לבעיית רגרסיה?
א. Accuracy
ב. Euclidean Distance
ג.SAE

2. אילו שיטות שיערוך מיועדים לבעיות clustering?
א. WSSE
ב. precision
ג. SAE
ד. כל התשובות נכונות

3. אילו שיטות שיערוך מיועדים לבעיות סיווג?
א. WSSE
ב. Error-rate
ג. MSE

Validation

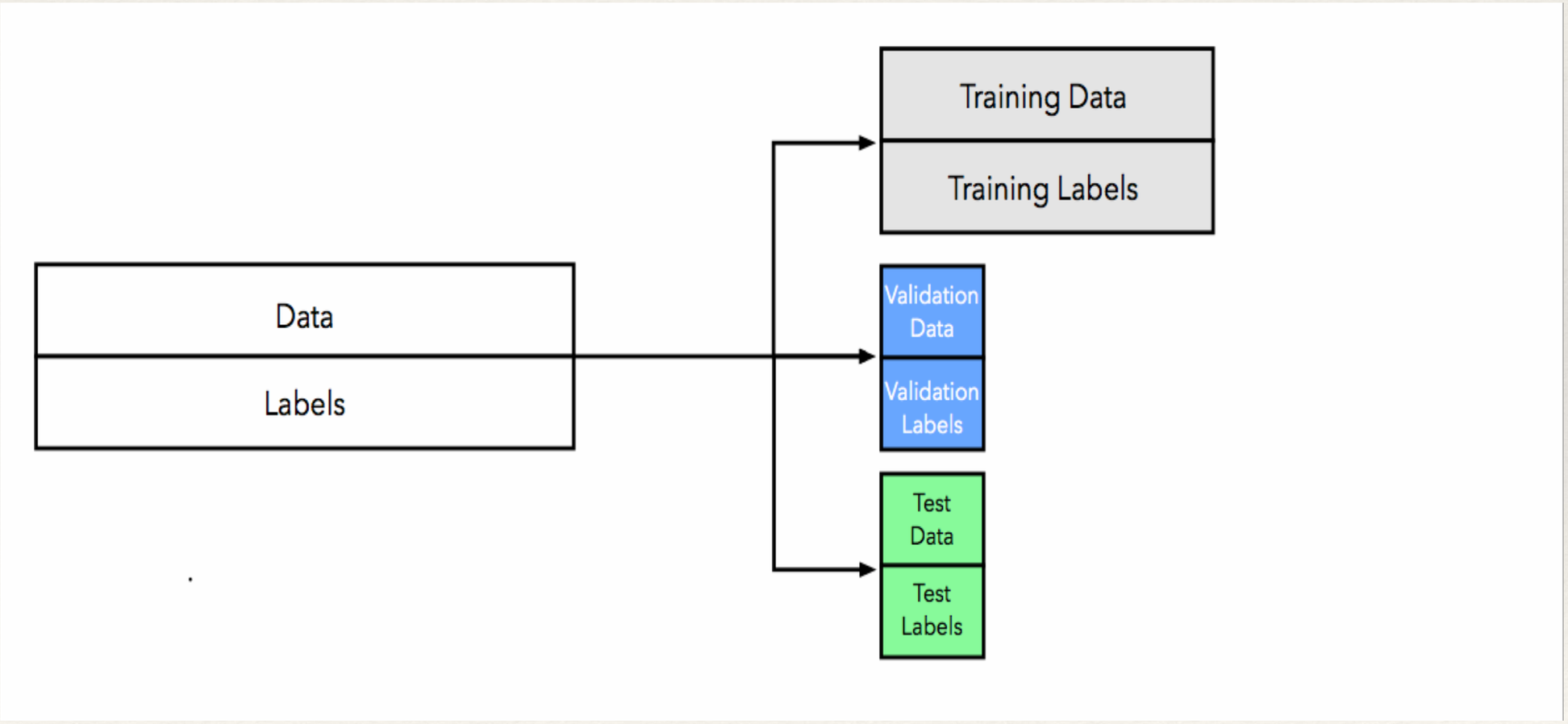
dataset – train-test-validation



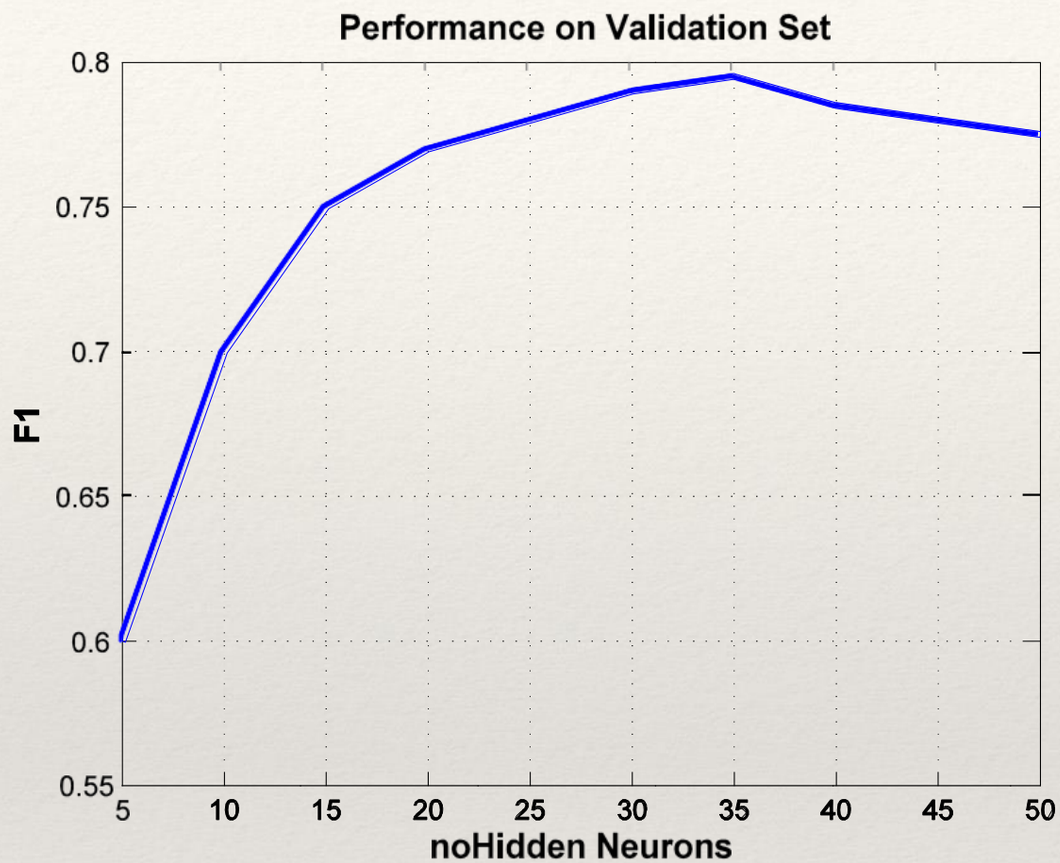
Holdout Method

- Split your dataset into 3 disjoint sets: Training, Validation, Test

- If a lot of data are available then you can try 50:25:25 otherwise 60:20:20.

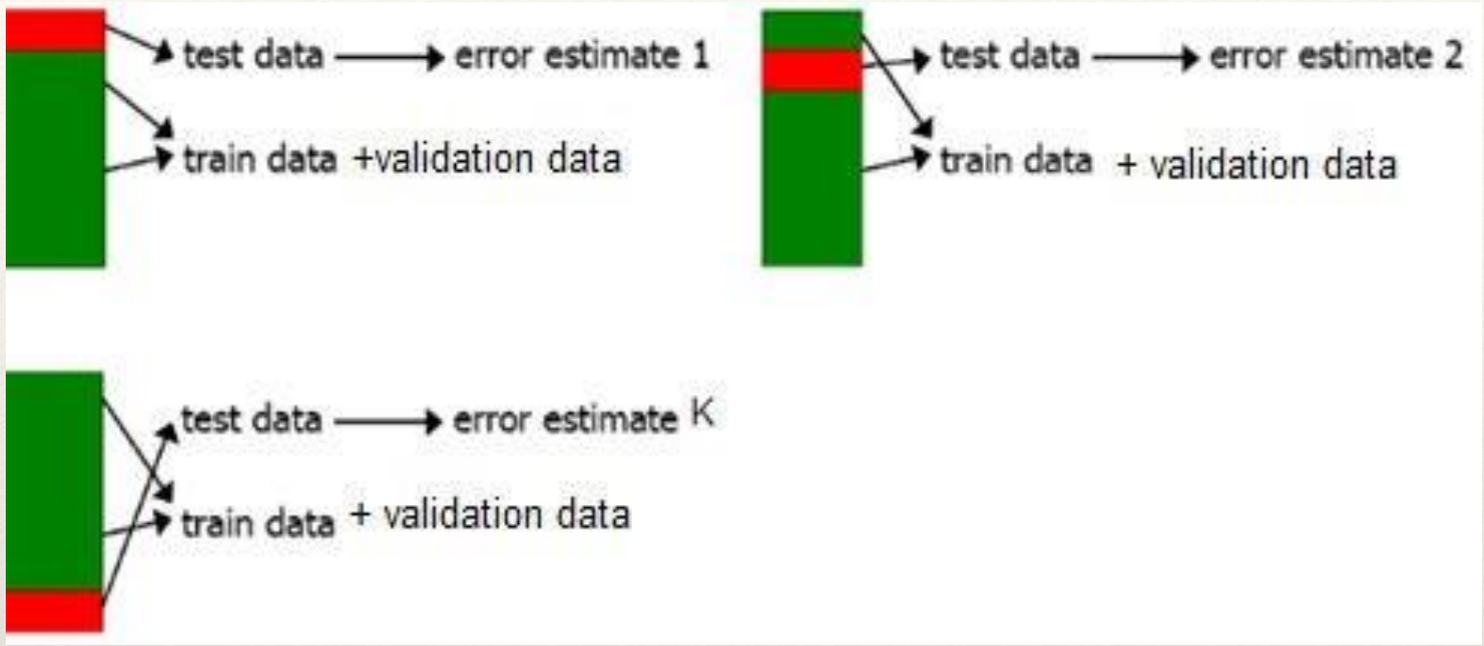


Holdout Method – Hyperparameter tuning



- Keep the classifier that leads to the maximum performance on the validation set (in this example the one trained with 35 hidden neurons).
- This is called parameter optimization/tuning, since you select the set of parameters that have produced the best classifier.

Cross Validation – Test Set Performance Estimation



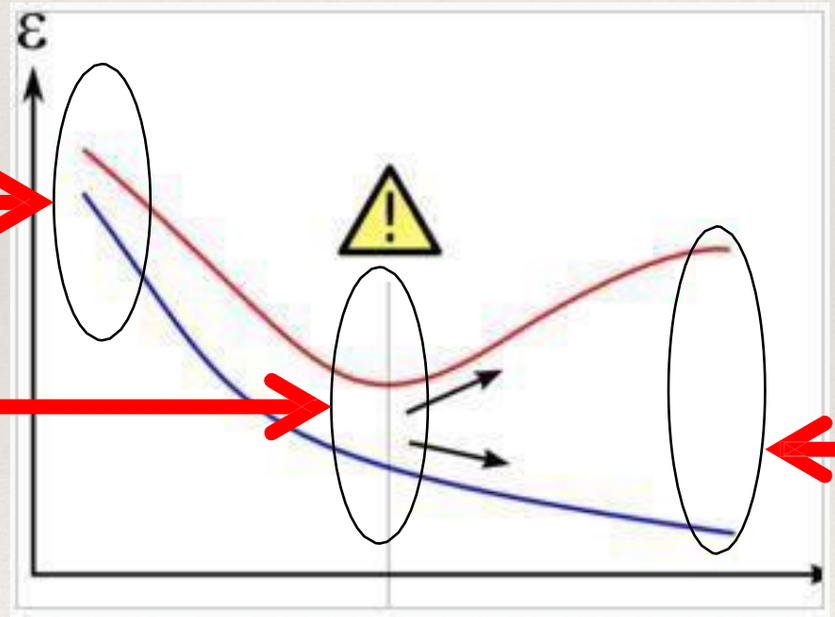
- Divide dataset into *k* (usually 10) folds using *k*-1 for training + validation and one for testing
- Test data between different folds should never overlap!
- Training + Validation and test data in the same iteration should never overlap!
- In each iteration the error on the left-out test set is estimated
- Error estimate: average of the *k* errors

Overfitting

- *Given a hypothesis space H, h∈H overfits the training data if there exists some alternative hypothesis h′∈ H such that h has smaller error than h′ over the training examples, but h′ has smaller error than h over the entire distribution of instances.*

Underfitting

Just right



- Red: error on Test set (unseen examples)
- Blue: error on Training set

Overfitting

- Overfitting: Small error on training set, but large error on unseen examples.
- Underfitting: Larger error on training and test sets.

שימושים ל validation set

- סיוע במניעת overfitting
- Model selection
- בחירת hyperparameters מיטביים
- תהליכים משלימים לתהליך האימון
  - Post pruning של עצי החלטה
  - בדיקה נקודה טובה לעצירה ב- ANN
- שיערוך המודל, בהיעדר test מקובל cross validation

ועוד ...

סילום (scaling)

מוטיבציה –

- למאפיינים שונים פונקציית התפלגות שונה
- KNN לא מניח איזשהם הנחות על התפלגות הנתונים (כנ"ל לאלגוריתמי clustering שלמדנו)
- סולם (scale) שונה עלול להוביל לעיוות המרחק ולכן מעוות גם את אלגוריתם k-means
- פגיעה גם ב-linear regression
- משפרת גם אלגוריתם רבים.

סילום (Scaling):

- standardization (t-distribution): נהפוך את התפלגות המאפיין ב-training להתפלגות t
- minmax normalization – שינוי הטווח לאחיד בין [0,1] ; [-1,1]

סילום - שאלות

שאלה 1: את מה משנה פעולת הסילום?

שאלה 2: איך מחשבים Standardization t-distribution?

שאלה 3: מה מבצע Min-max normalization?

שאלה 4: מתי מסוכן לבצע סילום?

דמיון ומרחק

Manhattan Distance: $d = \sum_{i=1}^{n} |x_i - y_i|$

Euclidean Distance: $d(x, y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$

Chebyshev Distance:

$$d(\overrightarrow{x_j}, \overrightarrow{x_i}) = \max_{1 \le m \le d} |x_{j_m} - x_{i_m}|$$

Cosine similarity : $d(\overrightarrow{x_j}, \overrightarrow{x_i}) = \frac{\overrightarrow{x_j}^T \cdot \overrightarrow{x_i}}{\|\overrightarrow{x_j}\| \cdot \|\overrightarrow{x_i}\|}$

Edit distance

שאלות ביניים

שאלה 1: מה הקשר בין מרחק, קרבה ושכנות? מה הקשר לקביעת הקטגוריה של הדוגמה החדשה (נניח בדוגמה של השכן הקרוב ביותר)?

שאלה 2: האם יכול להיות שמרחק צ'בישב גדול ממרחק מנהטן?

שאלה 3: מה הקשר בין מרחק אוקלידי למרחק בין נקודות במרחב?

שאלה 4: כיצד משתמשים במרחק או בדמיון ב- k-means

שאלה 5: מה הקשר בין פונקציות מרחק לשערוך מודל רגרסיה?

הקשרים שונים – Information Theory

- Entropy $H(X) = -p_1 \log_2 p_1 - p_2 \log_2 p_2 - \ldots - p_n \log_2 p_n$

$$= -\sum_{j=1}^{n} p_j \log_2 p_j$$

- Information Gain: $Gain(Y \mid X) = H(Y) - H(Y \mid X)$

$$\mathrm{NMI} = \frac{I(f1; f2)}{|H(f1) + H(f2)|/2}$$

מודלים הסתברותיים

- The Naïve Bayes classifier
  - Conditional independence assumption
  - How to train it?
  - How to make predictions?
  - How does it relate to other classifiers we know?

- Fundamental Machine Learning concepts
  - Bayes optimal classifier
  - Maximum Likelihood estimation
  - Generative story

Bayes' Rule

Class observation

$$\boxed{p(c \mid x) = \frac{P(x \mid c)P(c)}{P(x)}}$$

P(c) – ***prior*** probability of class *c* before any vector is seen

P(x|c) – ***likelihood*** of the observed data if the class is *c*

P(x) – ***evidence*** probability of the data

P(c|x) – ***posterior*** Probability of class c after the data is seen

Naïve Bayes Classifier

- Using Bayes rule:

$$P(c \mid x_1, x_2, \ldots\ldots x_D) = \frac{P(C)P(x_1, x_2, \ldots\ldots x_D \mid C)}{P(x_1, x_2, \ldots\ldots x_D)}$$

- Select the feature set such that each feature x_i is independent of every other feature x_j.

$$P(x_1, x_2, \ldots\ldots x_D \mid c) = P(x_1 \mid c)P(x_2 \mid c)P(x_3 \mid c)\ldots P(x_D \mid c) = \prod_{i=1}^{D} P(x_i \mid c)$$

שאלות – Naïve Bayes

שאלה 1: מה הבדל בין prior probability לבין posterior probability?

שאלה 2: מה הבדל בין מודל גנרטיבי למודל דיסקרמנטיבי? לאיזה סוג משתייך מודל של Naïve Bayes?

שאלה 3: מהי ההנחה הנאיבית ב-Naïve Bayes?

שאלה 4: את מה מנסים לפתור בעזרת smoothing?

K-NN

Most basic learning method

- Distance based classification
- Scaling is key here
- Hyper parameter tuning
- Several optimization methods
- No real training here..
- Much much more....

שאלות - KNN

שאלה 1:
מיהם השכנים? שכנים של מי? מה הקשר בין ה-training-set ל-test-set ב-KNN?

שאלה 2: מהם ה-hyper parameters שאפשר לעשות עבורם tuning ב-knn? ואיך עושים להם tuning?

שאלה 3:
כיצד נקבל החלטה לגבי הקטגוריה של דוגמה חדשה ע"י KNN? מה פירוש 1-NN בעצם? מה ההבדל בין בחירה לפי הרוב, לבין בחירה ממושקלת?

שאלה 4:
מה המשמעות של ערכי K שונים (קטנים וגדולים)? מדוע נשאף לבחור K אי זוגי אם מספר הקטגוריות הוא 2?

עצי החלטה

Intuitive learning methods

- Self explainable
- Information theory (where else do we use it?)
- Evaluation methods, confusion matrix
- Overfitting risks
- How to handle overfit
- Discrete attributes
- Supports mutli class (number of classes > 2)
  - As some of the other algorithms we've learned

עצי החלטה ותורת האינפורמציה - שאלות

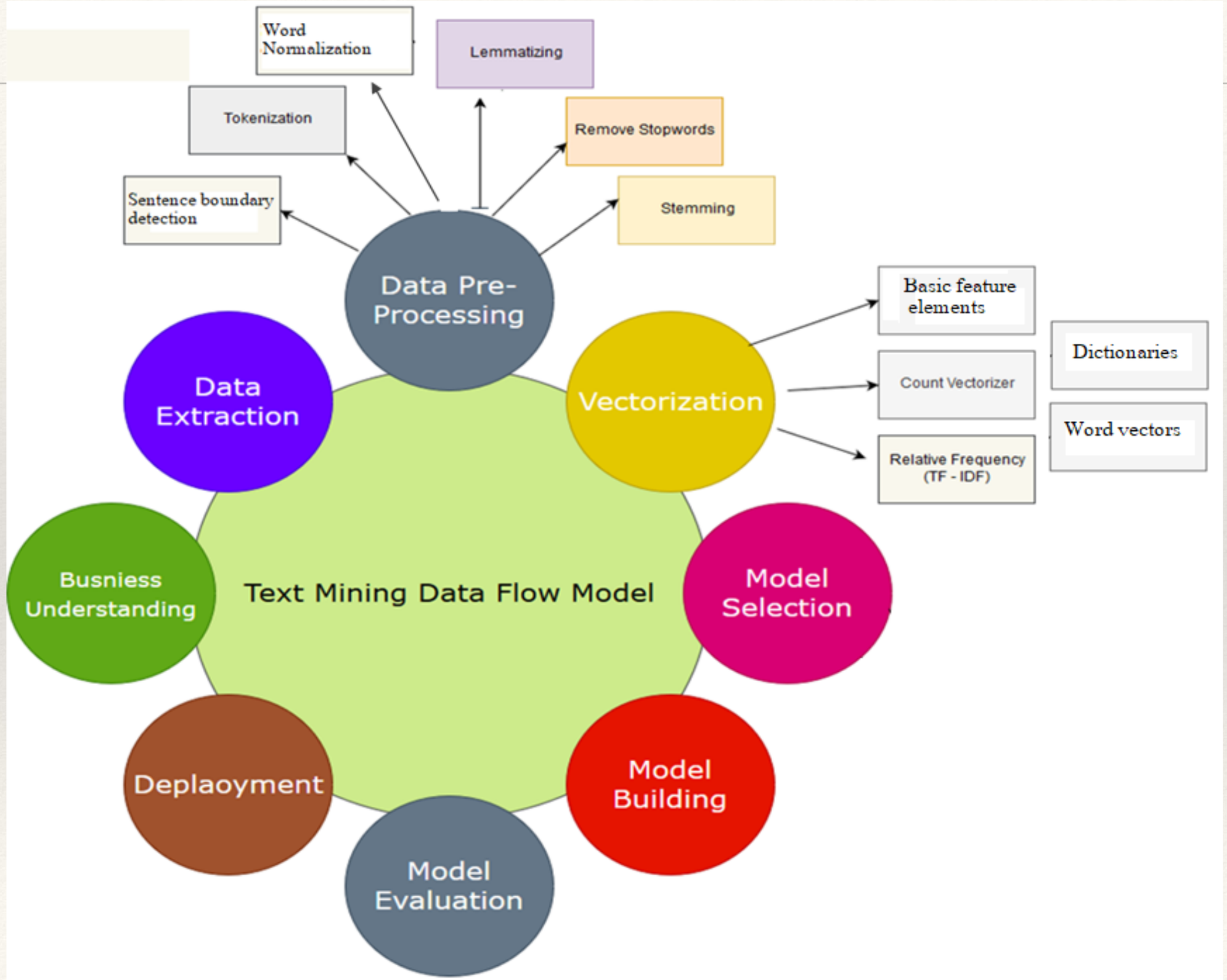
שאלה 1: ממה מורכבים עצי החלטה ומה מייצגים הרכיבים השונים בעץ? ומה הקשר בינהם בין ה-feature-set, ה-feature-vector והקטגוריות? ?

שאלה 2: האם נעדיף הפרדה "מסובכת" או "פשוטה" בין המחלקות? מה היתרון והחסרון של על אפשרות? ומה מנחה אותנו בעצי החלטות?

שאלה 3: כדי למצוא את המאפיין הבא בעץ, האם נחפש מאפיין שיגדיל את האנטרופיה בכל תת קבוצה של העץ או יקטין אותה?

שאלה 4: היכן עוד למדנו שאפשר להשתמש בתורת האינפורמציה בלמידת מכונה?

Basic Text Analysis Flow



Vectorization: extracting basic feature units

The bag of words (BOW) model:

Each **word** is treated as a feature in a unit called **document**.

Each such word will become a feature

- Alternative: **original tokens – no processing**
- Alternative: **normalized words – e.g., lemmas, stems**
- Alternative: **characters – e.g., prefixes**
- Alternative: **ngrams – unigram, bigram, trigram**
- More complex alternatives …

Vectorization: feature's value

Count Vectorizer - Converts a collection of text documents to a matrix of token counts

The bag of words (BOW) model:

Word Count

- Binary, $f_{t,d}$, $\mathrm{tf}(t,d) = \frac{f_{t,d}}{\sum_{t' \in d} f_{t',d}}$, $\log(1 + f_{t,d})$

How rare is the word?

inverse document frequency (idf) = $\log \frac{N}{|\{d \in D : t \in d\}|}$

number of documents

number of documents containing term

$$\mathrm{tfidf}(t,d,D) = \mathrm{tf}(t,d) \cdot \mathrm{idf}(t,D)$$

Tf·idf Vectorizer - Converts a collection of raw documents to a matrix of TF-IDF features

מודלים לינארים

- What are linear models?
  - a general framework for binary classification
  - how optimization objectives are defined
- loss functions
  - separate model definition from training algorithm (Gradient Descent)

מסווג לינארי – שאלות בסיסיות

1. איזו משוואה דיסקרמינטיבית (מפרידה) מהמשוואות הבאות הינה משוואה לינארית?

א. $3x_1+x_2+5=0$ ב. $x_1^2+2x_2+1=0$

2. עבור משוואה $3x_1+x_2+5=0$, מה יהיה הסיווג של זוגות הערכים הבאים (האפשרויות: חיובית/ שלילית)?

3. כיצד המשוואה הנ"ל עבור מודל רגרסיה, תחזה ערך עבור הוקטור (1,-1)?

Gradient Descent

- Gradient descent
  - a generic algorithm to minimize objective functions
  - what are the properties of the objectives for which it works well?
  - subgradient descent (ie what to do at points where derivative is not defined)
  - why choice of step size, initialization matter

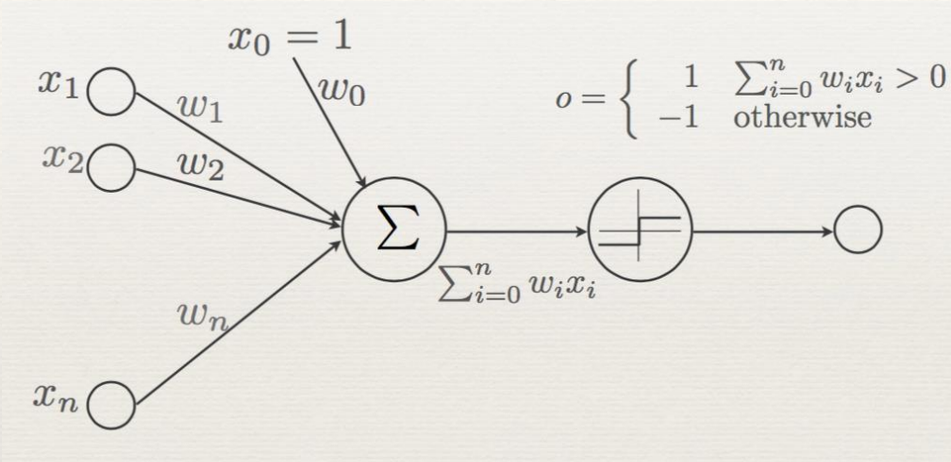
- מהי פונקצית גרדיאנט? מה הקשר שלה ללמידת מכונה?
- למה זקוקים לאלגורתם Gradient descent?

רשתות נוירונים

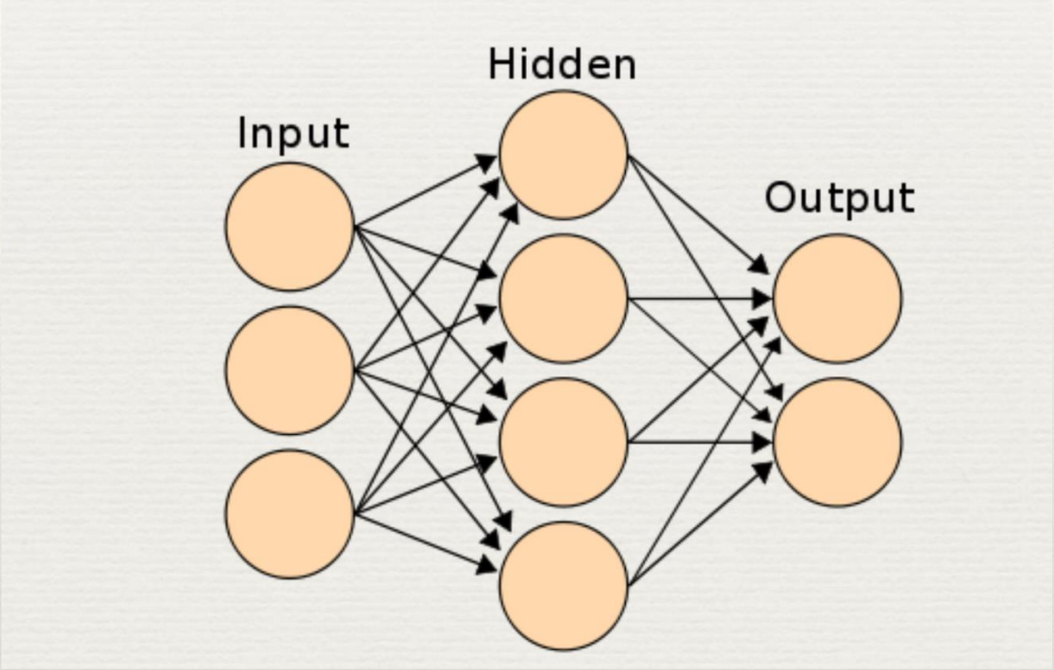
- What are Neural Networks?
  - Multilayer perceptron
- How to make a prediction given an input?
  - Forward propagation: Matrix operations + non-linearities
- Why are neural networks powerful?
  - Universal function approximators!
- How to train neural networks?
  - The backpropagation algorithm
    - How to step through it, and how to derive update rules

ANN

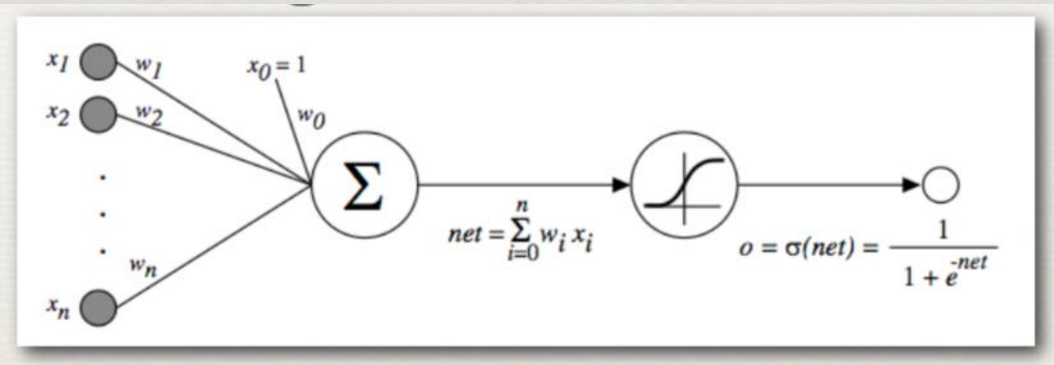
Perceptron w/ signum



Multilayer



Neuron w/ sigmoid



$\sigma(x)$ is the sigmoid function

$$\frac{1}{1 + e^{-x}}$$

$$\frac{d\sigma(x)}{dx} = \sigma(x)(1 - \sigma(x))$$

Backpropagation –

forward (result) → Backward (error)

שאלות - ANN

1. מהי פונקציית אקטיבציה?

2. מהו perceptron?

3. כיצד מתקנים את המשקולות באלגוריתם BACKPROPAGATION?

4. מהם ה-hyper parameters שלמדנו עבור ANN?

עיבוד תמונה - דוגמה לקונבולוציה של גרדיינט אנכי

3	0	1	2	7	4
1	5	8	9	3	1
2	7	2	5	1	3
0	1	3	1	7	8
4	2	1	6	2	8
2	4	5	2	3	9

6 X 6 image

*

1	0	-1
1	0	-1
1	0	-1

3 X 3 filter

<u>האלמנט הראשון</u> שמכפילים (מהמטריצה המקורית):

3^{1}	0^{0}	1^{-1}
1^{1}	5^{0}	8^{-1}
2^{1}	7^{0}	2^{-1}

<u>תוצאת המכפלה</u>:

$$3*1 + 0 + 1*-1 + 1*1 + 5*0 + 8*-1 + 2*1 + 7*0 + 2*-1 = -5$$

עיבוד תמונה - דוגמה לקונבולוציה של גרדיינט אנכי

10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0

*

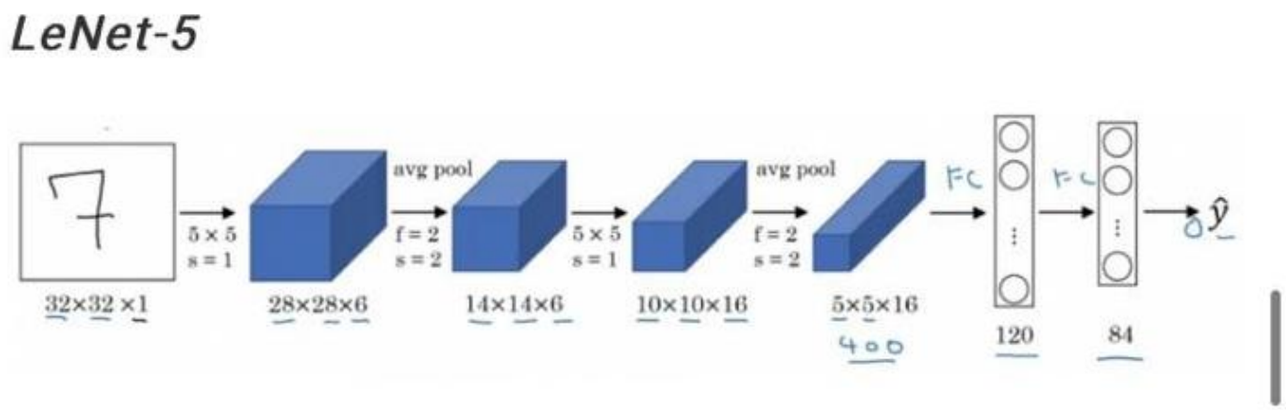
1	0	-1
1	0	-1
1	0	-1

=

0	30	30	0
0	30	30	0
0	30	30	0
0	30	30	0

ניתן לראות שהקונוולוציה משמשת לגילוי קצוות

עיבוד תמונה - דוגמה לרשת LeNet-5



It takes a grayscale image as input. Once we pass it through a combination of convolution and pooling layers, the output will be passed through fully connected layers and classified into corresponding classes. The total number of parameters in LeNet-5 are:

- **Parameters:** 60k
- **Layers flow:** Conv -> Pool -> Conv -> Pool -> FC -> FC -> Output
- **Activation functions:** Sigmoid/tanh and ReLu

סוגי בעיות אופטימיזציה

Convex programming - בעיות אופטימיזציה של מזעור, בהם פונקצית המטרה הם קמורות (Convex).

Quadratic programming – בבעיות אופטימיזציה – מאפשר לביטוי המופיע בפונקצית המטרה (אותה רוצים למקסם או למזער) להיות ביטוי ריבועי (quadratic)

- דוגמה לכך, נראה בפונקצית אופטימיזציה של SVM.
- תזכורת – ב-SVM המטרה ליצור מרווח (margin) מקסימלי בין שתי המחלקות

הערה: כמובן, ישנם סוגים נוספים שונים של פתרונות ובעיות אופטימיזציה

בעיית סיפוק אילוצים (constraint satisfaction problem)

בעיות סיפוק אילוצים הן בעיות של השמת ערכים למשתנים כך שיש אילוצים מסוימים בין ערכים

נתייחס ל-2 סוגי אילוצים אפשריים:

- אילוצי שיוויון (equality)
- אילוצי השיוויון נראים כך:

$$g_i(\mathbf{x}) = c_i \quad \text{for } i = 1, \ldots, n$$

- אילוצי אי-שיוויון (inequality)
- אילוצי אי-השיוויון נראים כך:

$$h_j(\mathbf{x}) \geq d_j \quad \text{for } j = 1, \ldots, m$$

constraint optimization problems – תת סוג של בעיות סיפוק אילוצים, בהם האילוץ אינו קשיח, ובעצם המטרה, היא להוריד את מחיר האילוצים למינימום.

- אנחנו נתייחס *constraint optimization problems* ב-SVM

למידת מכונה – כפונק' קירוב

Problem setting

- Set of possible instances X
- Unknown target function $f: X \rightarrow Y$
- Set of function hypotheses $H = \{h \mid h: X \rightarrow Y\}$

Input

- Training examples $\{(x^{(1)}, y^{(1)}), \ldots (x^{(N)}, y^{(N)})\}$ of unknown target function f

Output

- Hypothesis $h \in H$ that best approximates target function f

SVM

- What are Support Vector Machines
  - Hard margin vs. soft margin SVMs
- How to train SVMs
  - Which optimization problem we need to solve
- Geometric interpretation
  - What are support vectors and what is their relation with parameters **w**,b?
- How do SVM relate to the general formulation of linear classifiers
- Why/how can SVMs be kernelized
- Kernel functions
  - What they are, why they are useful, how they relate to SVM?
  - Where else could we use kernels?

איך מחשבים את ה-margin

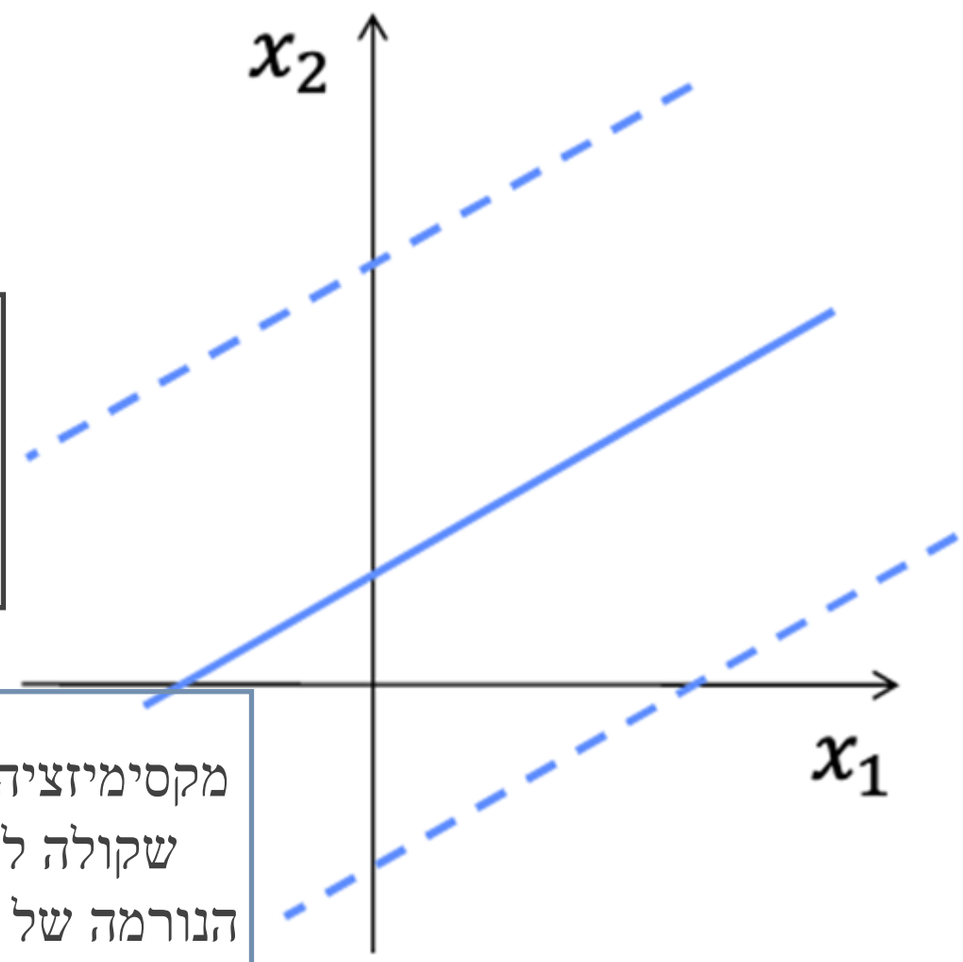
Point-plain distances from the two margins to the origin:

$$d_+ = \frac{\left\|\left(w\cdot 0\right)+b+1\right\|}{\|w\|}, d_- = \frac{\left\|\left(w\cdot 0\right)+b-1\right\|}{\|w\|}$$

מודדים את המרחק מהקו העליו לראשית הצירים ובאופן דומה לגבי הקו התחתון

$$\Rightarrow \mathrm{M} = \frac{2}{\|w\|}$$

חישוב ה-margin



Objective:

$$max(\mathrm{M}) \rightarrow min(\|w\|^2)$$

$$s.t.$$

מקסימיזציה של ה-margin שקולה למינימיזציה של הנורמה של וקטור המשקולות

חישוב w

Calculating w, b:

$$w = \sum_i \alpha_i y_i x_i$$

קבועי לגראנג' - $\alpha_i \geq 0$ – אם מדובר ב-support vectors, אחרת שווים ל-0

$$y_i\left(w \cdot x_i + b\right) - 1 \geq 0 \ \forall i$$

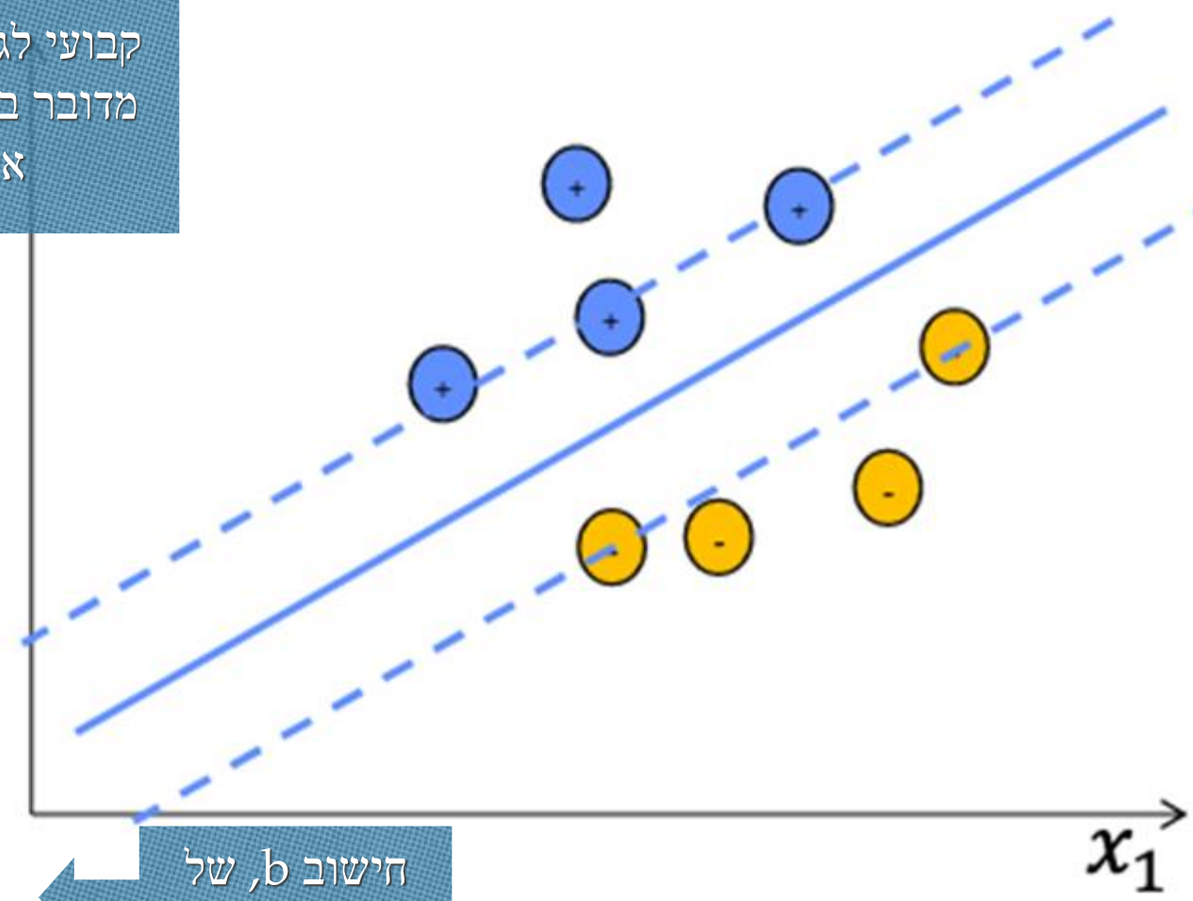
$$b_+ = \min\left(b_i\right) \ ; y_i = +1$$

$$b_- = \max\left(b_i\right) \ ; y_i = -1$$

$$b = \frac{b_+ + b_-}{2}$$

$$b = -\frac{max_{y_i=-1}(\mathbf{w} \cdot \mathbf{x}_i) + min_{y_i=1}(\mathbf{w} \cdot \mathbf{x}_i)}{2}$$

חישוב b, של הקו האמצעי

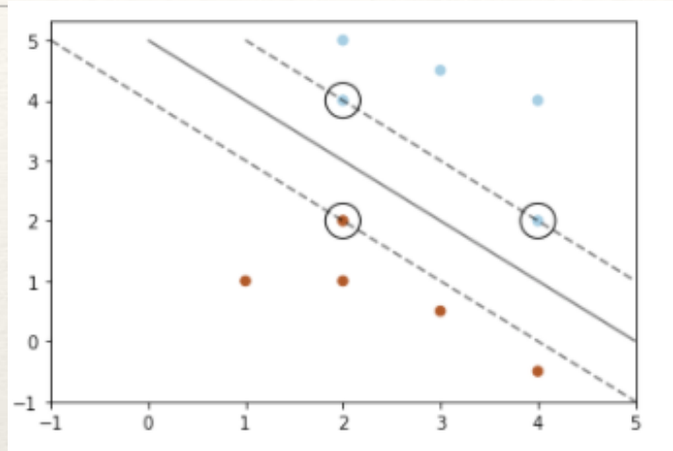


Final decision function:

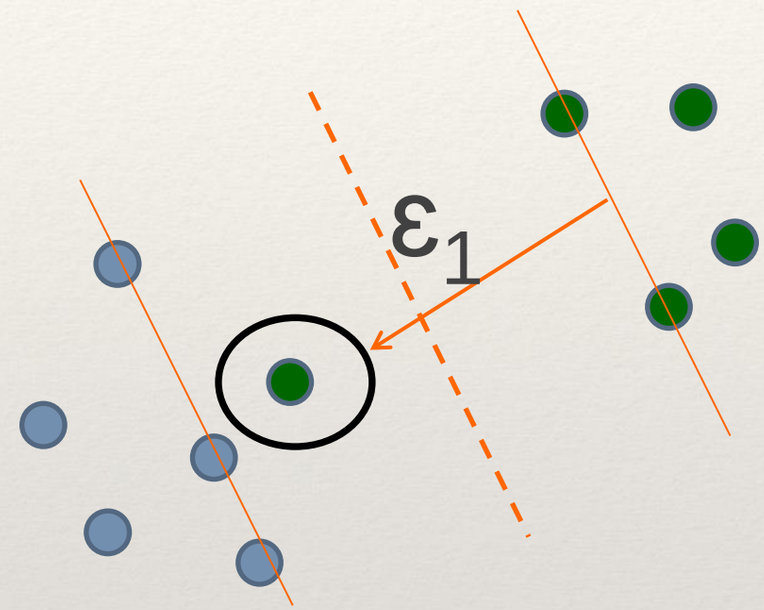
$$f\left(x\right) = sign\left(w \cdot x + b\right) = sign\left(\sum_{i=1}^{l} \alpha_i y_i x_i \cdot x + b\right)$$

החלטת סיווג

SVM - שאלות



1. כמה וקטורים תומכים יש בדוגמא משמאל?



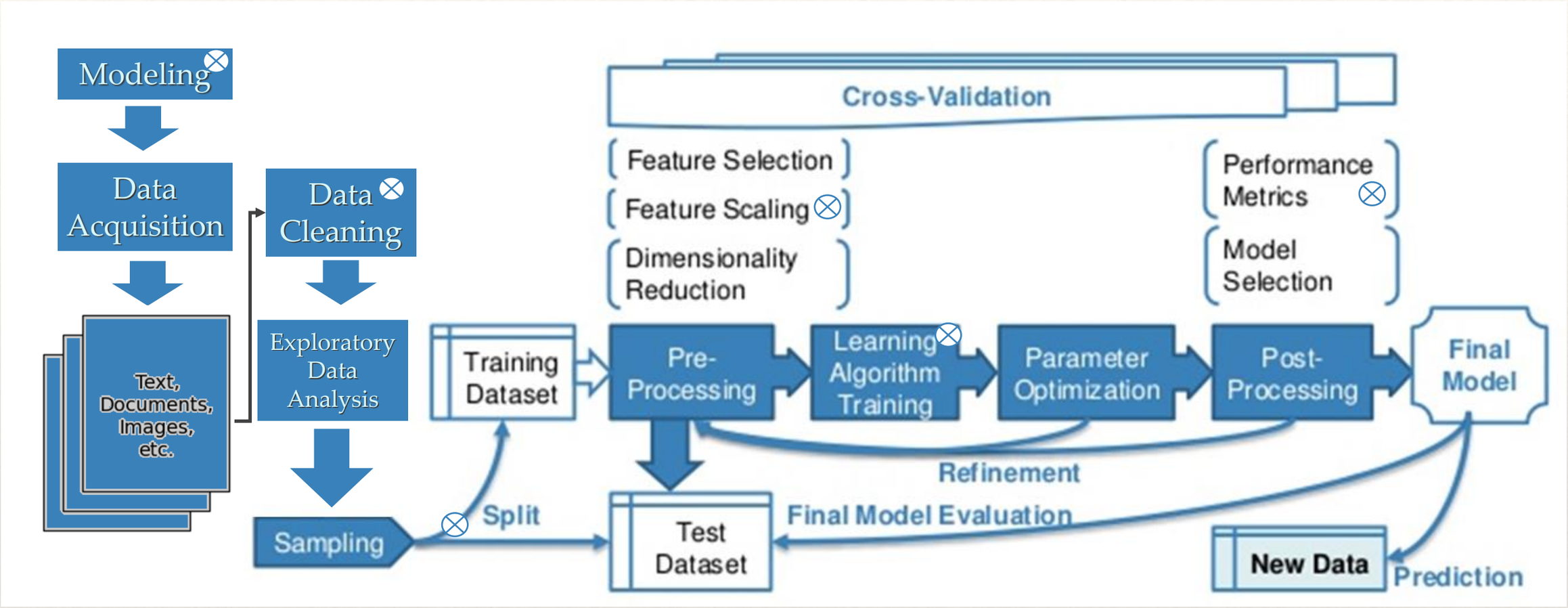
2. האם הנקודה המוקפת בעיגול (משמאל) הינה וקטור תומך או שגיאה?

3. מהם קבועי לאגרנג' ולמה הם משמשים (2 שימושים)?

4. מה משמשים kernels?

5. האם עדיף margine מינימלי או מקסימלי?

A typical regression flow - diving in



Data Cleaning

→ Duplicates

→ Missing Data

→Remove

→Repair

Train-Test split

+ sampling

Scaling

→ Minmax norm.

→ t-dist. standardization

Feature Selection

→ Feature compared to itself.

→ compared to other features.

Learning Algos.

→ Linear regression

Validation

→ Hyper parameter tuning

Post Processing

Evaluation

→ SAE

→ MAE

→ SSE

→ MSE

→ RMSE

→ R^2

רגרסיה לינארית - פונקצית מחיר (Cost Function)

המודל הלינארי: $\hat{y}_i = \vec{w} \cdot \vec{x}_i$

פונקצית המחיר

$$J(\vec{w}) = \frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2 = \frac{1}{n}\sum_{i=1}^{n}(\vec{w}\cdot\vec{x}_i - y_i)^2$$

$(n \geq 1)$

Multivariate Gradient Descent Algorithm - for linear regression:

Repeat until done:

We want w_0 to be partialy derived as the rest of $\vec{w}$, so if j=0, $x_{i,0} = 1$

$$w_j = w_j - \alpha \cdot \frac{\partial J}{\partial w_j} = w_j - \alpha \cdot \frac{2}{n} \cdot \sum_{i=1}^{n}\left[(\vec{w}\cdot\vec{x}_i - y_i)\cdot x_{i,j}\right]$$

(simultaneously update w_j for j=0,…,d)

רגרסיה לינארית - שאלות

1. בפונקציה רציפה וגזירה פעמיים, כיצד נמצא את נקודת המינימום? א. אם הפונקציה, בעל נגזרת ראשונה >0, לפני הנקודה, ונגזרת שנייה שווה לאפס, זוהי נקודת מינימום.
ב. אם הפונקציה בעלת ערך=0 בנגזרת הראשונה בנקודה ונגזרת שניה>0, זוהי נקודת מינימום.

2. לאיזה כיוון נצטרך להתקדם, בשביל להתקרב למינימום בפונקציה?

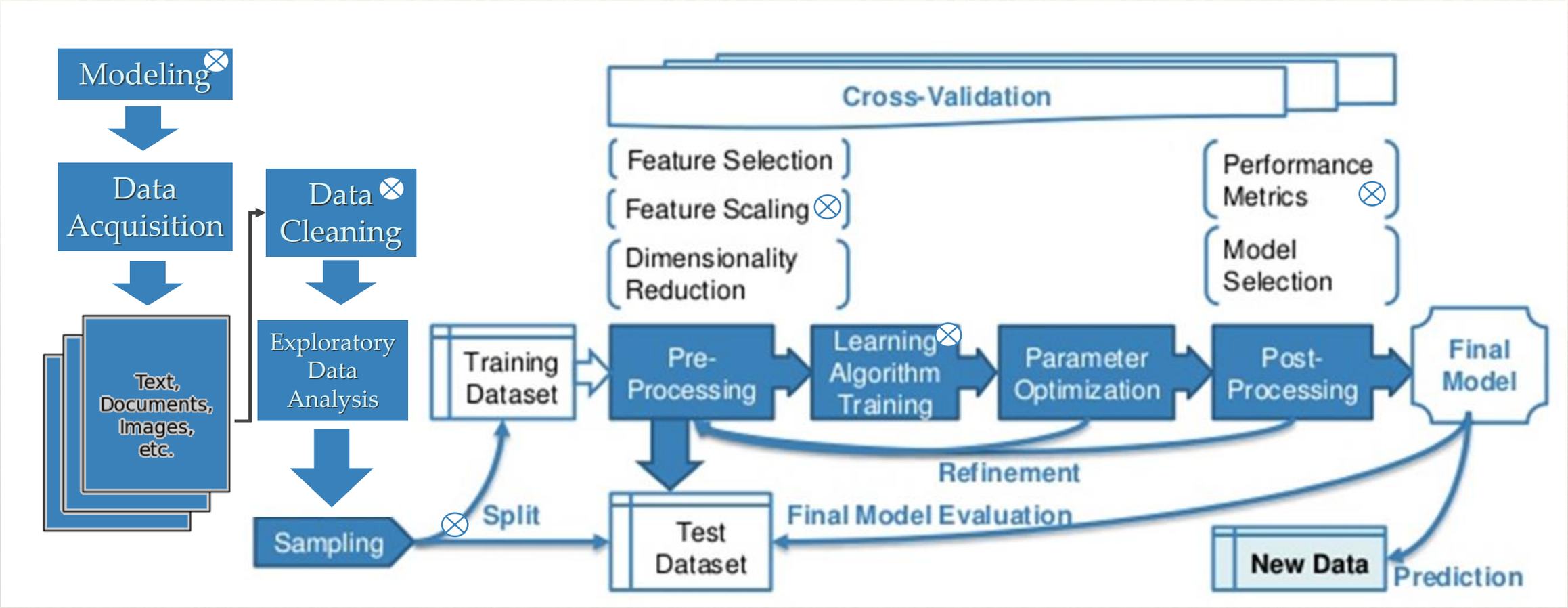
3. מהי פונקציה קמורה (פונקציית convex)?

4. מהו גראדיאנט?

5. מהו מטריצת הסיאן ומדוע לא נשתמש בה לאימון מודל רגרסיה לינארית?

6. מה הקשר בין פונקצית הפסד לאימון רגרסיה לינארית?

A typical clustering flow - diving in



Data Cleaning

→ Duplicates

→ Missing Data

→Remove

→Repair

Scaling

→ Minmax norm.

→ t-dist. standardization

Feature Selection

→ Feature compared to itself.

→ compared to other features.

Learning Algos.

→ K-means

Extensions

Evaluation

→ WSSE (WSS)

Clustering

What are other unsupervised problems

Why is clustering difficult

K-means

Distance methods

Evaluation methods

Clustering

- What types of errors do we have re: clustering?
- Is k-means loss function a convex function?
- Where did we see convex function?
- How do we know we have good clusters?

שאלות – Clustering

1. איזו מהבעיות הבאות נרצה לפתור בעזרת clustering?
א. בניית מודל שיחליט האם תמונה מסויימת היא של הולך רגל או לא
ב. בניית מודל שימצא קבוצות חברים ברשת חברתית
ג. בניית מודל שיחזה את תחזית מזג האוויר מחר

2. באיזו שיטה ישוייך הוקטור רק ל-cluster אחד?

3. איך נחשב את ה-prototype לכל cluster ב-kmeans ואיך נדע שה-cluster אייכותי ביחס לווקטרים השייכים אליו?

שאלות הבנה לדוגמא

- After training a SVM, we can discard all examples which are not support vectors and can still classify new examples. True or False?

- In knn, if we increase the k hyper-parameter, does it increase or decrease the variance?

דברים שלא למדנו (ויש המון..)

Advanced generalization bounds:

- Stability based
- More Radamacher
- PAC Bayes.
- Compression bounds

דברים שלא למדנו (ויש המון..)

Optimization for machine learning:

- Coordinate descent
- Adaptive gradients
- Variance reduction methods

דברים שלא למדנו (ויש המון..)

Deep learning

- Theory of non-convex optimization
- Sequence to sequence models
- Unsupervised deep learning
- Getting it to work!

דברים שלא למדנו (ויש המון..)

Reinforcement learning (RL)

- Humans behave in an environment, and our actions
- affect and are affected by our surroundings.
- The RL framework nicely captures this and is in a sense the most general learning setting.
- Much recent progress when combined with deep learning
- Much much more....

למידת מכונה – סיכום סמסטר

- אלגוריתמי הלמידה שלמדנו:
  - K-NN
  - עצי החלטה
  - Naive Bayes
  - perceptrons
  - רשתות נוירונים
  - SVM
  - רגרסיה לינארית
  - PCA
  - K-means

בהצלחה לכולם ☺