7. Supervised $\rightarrow$ Unsupervised Techniques (flipped)

DS-GA 1015, Text as Data

Arthur Spirling

March 23, 2021

Housekeeping

Housekeeping

1. HW 2 coming in next week.

Housekeeping

1. HW 2 coming in next week.
2. OH will run 11-12 tomorrow (I have the general DGS meeting)

Housekeeping

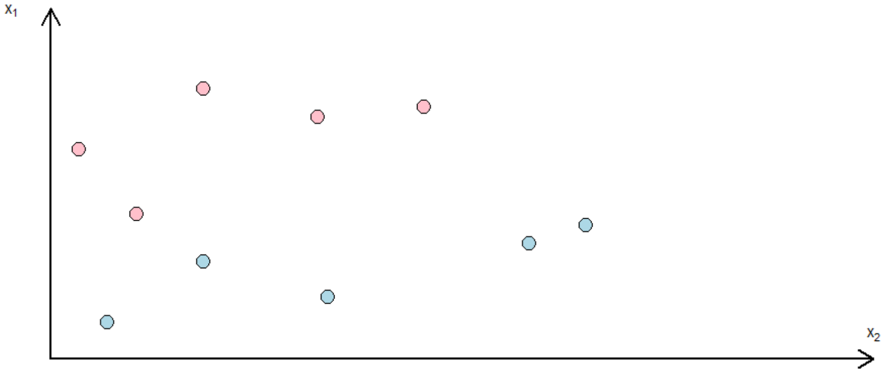
1 HW 2 coming in next week.

2 OH will run 11-12 tomorrow (I have the general DGS meeting)

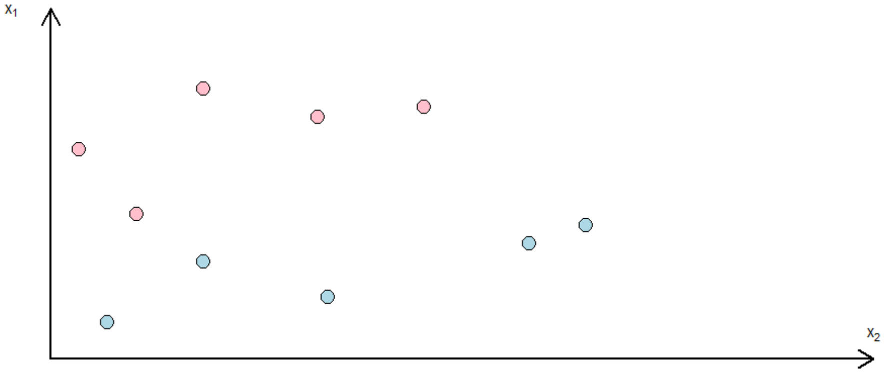
3 Lab at 12 today (no lab on Thursday)

Reminder: The 10 Senators

Reminder: The 10 Senators



Reminder: The 10 Senators



k-nearest neighbors

k-nearest neighbors

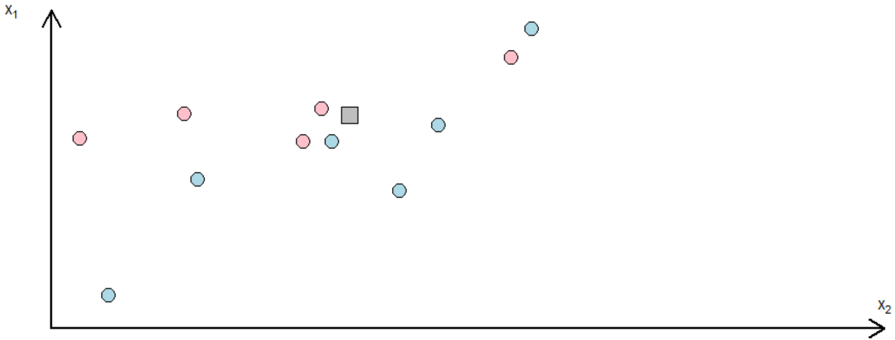
Variant of the Senate example:

k-nearest neighbors

Variant of the Senate example: now suppose we are interested in classifying a 'new' (test set) Senator (■) based on her feature values.

k-nearest neighbors

Variant of the Senate example: now suppose we are interested in classifying a 'new' (test set) Senator (■) based on her feature values.


x_1
x_2

k-nearest neighbors

Variant of the Senate example: now suppose we interested in classifying a 'new' (test set) Senator (■) based on her feature values.

k-nearest neighbors

Variant of the Senate example: now suppose we interested in classifying a 'new' (test set) Senator (■) based on her feature values.

One simple idea is to look at her k nearest neighbors from the training set in the feature space,

k-nearest neighbors

Variant of the Senate example: now suppose we interested in classifying a 'new' (test set) Senator (■) based on her feature values.

One simple idea is to look at her k nearest neighbors from the training set in the feature space, and take a majority vote in terms of what party we should assign her to.

k-nearest neighbors

Variant of the Senate example: now suppose we interested in classifying a 'new' (test set) Senator (■) based on her feature values.

One simple idea is to look at her k nearest neighbors from the training set in the feature space, and take a majority vote in terms of what party we should assign her to.

e.g. $k = 3$:

k-nearest neighbors

Variant of the Senate example: now suppose we interested in classifying a 'new' (test set) Senator (■) based on her feature values.

One simple idea is to look at her k nearest neighbors from the training set in the feature space, and take a majority vote in terms of what party we should assign her to.

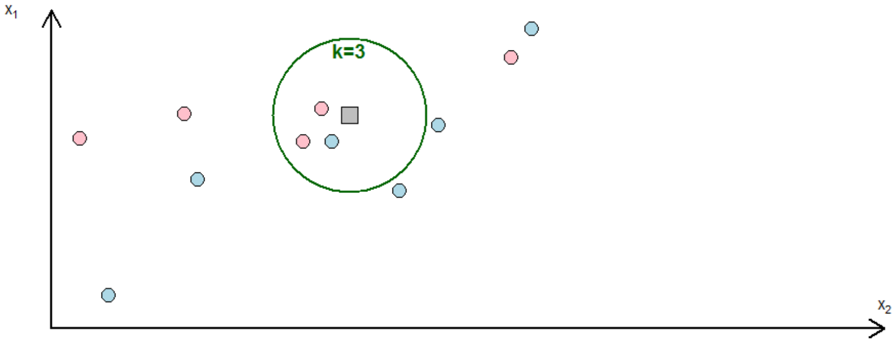
e.g. $k = 3$: she is assigned to the Republicans (2 vs 1)

k-nearest neighbors

Variant of the Senate example: now suppose we interested in classifying a 'new' (test set) Senator (■) based on her feature values.

One simple idea is to look at her k nearest neighbors from the training set in the feature space, and take a majority vote in terms of what party we should assign her to.

e.g. $k = 3$: she is assigned to the Republicans (2 vs 1)


x1
k=3
x2

k-nearest neighbors

Variant of the Senate example: now suppose we interested in classifying a 'new' (test set) Senator (■) based on her feature values.

One simple idea is to look at her k nearest neighbors from the training set in the feature space, and take a majority vote in terms of what party we should assign her to.

e.g. $k = 3$: she is assigned to the Republicans (2 vs 1)

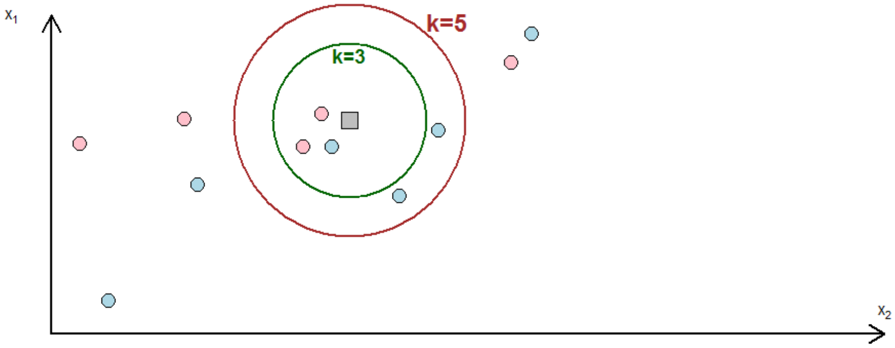
k-nearest neighbors

Variant of the Senate example: now suppose we interested in classifying a 'new' (test set) Senator (■) based on her feature values.

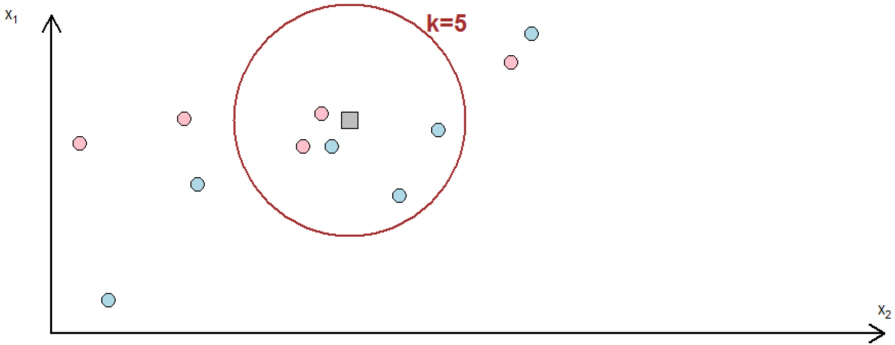
One simple idea is to look at her k nearest neighbors from the training set in the feature space, and take a majority vote in terms of what party we should assign her to.

e.g. $k = 3$: she is assigned to the Republicans (2 vs 1)

e.g. $k = 5$: she is assigned to the Democrats (3 vs 2)



x_1
k=5
k=3
x_2


k=5
x_1
x_2

k-nearest neighbors

Variant of the Senate example: now suppose we interested in classifying a 'new' (test set) Senator (■) based on her feature values.

One simple idea is to look at her k nearest neighbors from the training set in the feature space, and take a majority vote in terms of what party we should assign her to.

e.g. $k = 3$: she is assigned to the Republicans (2 vs 1)

e.g. $k = 5$: she is assigned to the Democrats (3 vs 2)

k-nearest neighbors

Variant of the Senate example: now suppose we interested in classifying a 'new' (test set) Senator (■) based on her feature values.

One simple idea is to look at her k nearest neighbors from the training set in the feature space, and take a majority vote in terms of what party we should assign her to.

e.g. $k = 3$: she is assigned to the Republicans (2 vs 1)

e.g. $k = 5$: she is assigned to the Democrats (3 vs 2)

→ Can use Euclidean distance (or some other metric for the neighbors), but need to be careful about imbalance in the training set.

k-nearest neighbors

Variant of the Senate example: now suppose we interested in classifying a 'new' (test set) Senator (■) based on her feature values.

One simple idea is to look at her k nearest neighbors from the training set in the feature space, and take a majority vote in terms of what party we should assign her to.

e.g. $k = 3$: she is assigned to the Republicans (2 vs 1)

e.g. $k = 5$: she is assigned to the Democrats (3 vs 2)

→ Can use Euclidean distance (or some other metric for the neighbors), but need to be careful about imbalance in the training set.

→ Choice of k can be optimized, but generally case that noise in data causes poor classification.

Recap Questions

Recap Questions

1 The k is usually odd. Why?

Recap Questions

1 The k is usually odd. Why?

2 We say kNN is lazy learner. Why?

Recap Questions

1 The k is usually odd. Why?

2 We say kNN is lazy learner. Why?

3 kNN is cheap to train but expensive to test. Why?

Recap Questions

1. The k is usually odd. Why?
2. We say kNN is lazy learner. Why?
3. kNN is cheap to train but expensive to test. Why?
4. kNN is non-parametric:

Recap Questions

1 The k is usually odd. Why?

2 We say kNN is lazy learner. Why?

3 kNN is cheap to train but expensive to test. Why?

4 kNN is non-parametric: what makes it so? what are some strengths/limitations of non-parametric models?

Exercise

Exercise

1. In practice, we have to be careful about using KNN techniques when there is imbalance in terms of proportions of classes in the training set. Why?

Exercise

1. In practice, we have to be careful about using KNN techniques when there is imbalance in terms of proportions of classes in the training set. Why? (hint: think about classifying a new point when 99.99% of all observations belong to one class.)

Exercise

1. In practice, we have to be careful about using KNN techniques when there is imbalance in terms of proportions of classes in the training set. Why? (hint: think about classifying a new point when 99.99% of all observations belong to one class.)

2. Suppose we have picked $k = 3$. In practice, we often weight the votes of the three nearest observations by $\frac{1}{d}$ where d is the distance from the observation we wish to classify. Why?

Exercise

1 In practice, we have to be careful about using KNN techniques when there is imbalance in terms of proportions of classes in the training set. Why? (hint: think about classifying a new point when 99.99% of all observations belong to one class.)

2 Suppose we have picked $k = 3$. In practice, we often weight the votes of the three nearest observations by $\frac{1}{d}$ where d is the distance from the observation we wish to classify. Why?

BTW imbalance

BTW imbalance

Having imbalanced data is a very common problem: often have very many more of one class than another (e.g. rare disease testing).

BTW imbalance

Having imbalanced data is a very common problem: often have very many more of one class than another (e.g. rare disease testing). We could...

BTW imbalance

Having imbalanced data is a very common problem: often have very many more of one class than another (e.g. rare disease testing). We could. . .

upsample minority class: randomly duplicate instances from minority class to boost their signal. Could resample with replacement.

BTW imbalance

Having imbalanced data is a very common problem: often have very many more of one class than another (e.g. rare disease testing). We could. . .

upsample minority class: randomly duplicate instances from minority class to boost their signal. Could resample with replacement.

downsample majority class: randomly remove instances from majority class to reduce their signal. Could sample m (number in minority class) without replacement.

BTW imbalance

Having imbalanced data is a very common problem: often have very many more of one class than another (e.g. rare disease testing). We could. . .

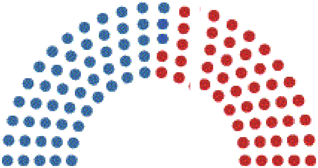
upsample minority class: randomly duplicate instances from minority class to boost their signal. Could resample with replacement.

downsample majority class: randomly remove instances from majority class to reduce their signal. Could sample m (number in minority class) without replacement.

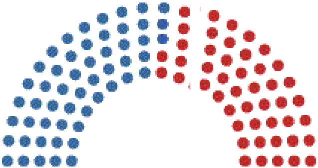
penalize mistakes in minority class: add a cost function.

Partitioning the Senators With Trees

Partitioning the Senators With Trees

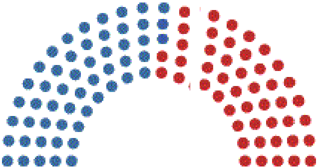


Partitioning the Senators With Trees



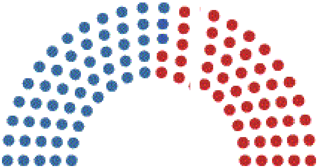
Idea our Senators are defined by their attributes.

Partitioning the Senators With Trees



Idea our Senators are defined by their attributes. Suppose we (optimally) split ('partition') the Senators with respect to x_1, such that we form two subsets of our training data.

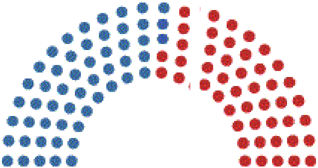
Partitioning the Senators With Trees



Idea our Senators are defined by their attributes. Suppose we (optimally) split ('partition') the Senators with respect to x_1, such that we form two subsets of our training data.

e.g suppose that Republicans generally use 'guns' more than Democrats,

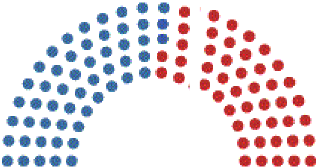
Partitioning the Senators With Trees



Idea our Senators are defined by their attributes. Suppose we (optimally) split ('partition') the Senators with respect to x_1, such that we form two subsets of our training data.

e.g suppose that Republicans generally use '`guns`' more than Democrats, such that grabbing all the observations for which $x_{guns} > 0.621$ captures, say, 80% of the Republicans in our data.

Partitioning the Senators With Trees

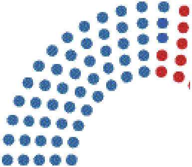


Idea our Senators are defined by their attributes. Suppose we (optimally) split ('partition') the Senators with respect to x_1, such that we form two subsets of our training data.

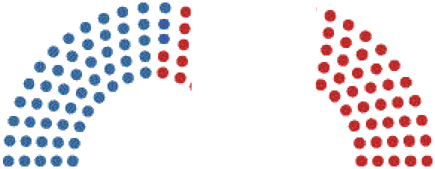
e.g suppose that Republicans generally use 'guns' more than Democrats, such that grabbing all the observations for which $x_{guns} > 0.621$ captures, say, 80% of the Republicans in our data.

Tree, stage 1

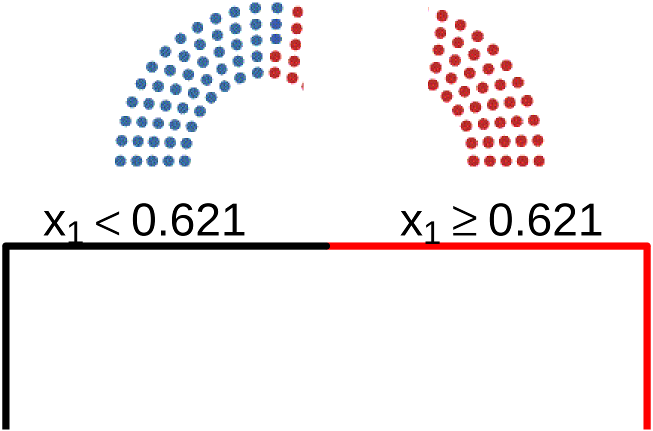
Tree, stage 1



Tree, stage 1

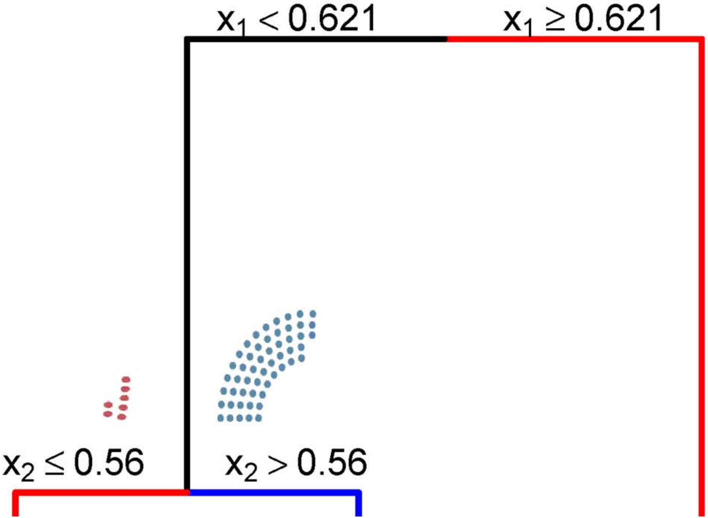


Tree, stage 1



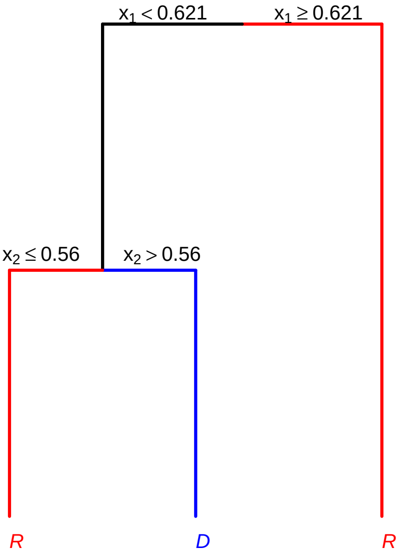
Tree, stage 2

Tree, stage 2



Complete Tree

Complete Tree



Recap Quiz

1 Some techniques grow multiple trees.

Recap Quiz

1 Some techniques grow multiple trees. Why? What can go wrong with one tree?

Recap Quiz

1. Some techniques grow multiple trees. Why? What can go wrong with one tree?
2. What is bagging?

Recap Quiz

1. Some techniques grow multiple trees. Why? What can go wrong with one tree?
2. What is bagging?
3. What is boosting?

Results of Hillard et al.

Results of Hillard et al.

TABLE 3. Bill Title Interannotator Agreement for Five Model Types

	SVM	MaxEnt	Boostexter	Naïve Bayes	Ensemble
Major topic $N = 20$	88.7% (.881)	86.5% (.859)	85.6% (.849)	81.4% (.805)	89.0% (.884)
Subtopic $N = 226$	81.0% (.800)	78.3% (.771)	73.6% (.722)	71.9% (.705)	81.0% (.800)

Note. Results are based on using approximately 187,000 human-labeled cases to train the classifier to predict approximately 187,000 other cases (that were also labeled by humans but not used for training). Agreement is computed by comparing the machine's prediction to the human assigned labels. (AC1 measure presented in parentheses).

Results of Hillard et al.

TABLE 3. Bill Title Interannotator Agreement for Five Model Types

	SVM	MaxEnt	Boostexter	Naïve Bayes	Ensemble
Major topic $N = 20$	88.7% (.881)	86.5% (.859)	85.6% (.849)	81.4% (.805)	89.0% (.884)
Subtopic $N = 226$	81.0% (.800)	78.3% (.771)	73.6% (.722)	71.9% (.705)	81.0% (.800)

Note. Results are based on using approximately 187,000 human-labeled cases to train the classifier to predict approximately 187,000 other cases (that were also labeled by humans but not used for training). Agreement is computed by comparing the machine's prediction to the human assigned labels. (AC1 measure presented in parentheses).

AC1 is intercoder reliability corrected for chance agreement.

Results of Hillard et al.

TABLE 3. Bill Title Interannotator Agreement for Five Model Types

	SVM	MaxEnt	Boostexter	Naïve Bayes	Ensemble
Major topic *N* = 20	88.7% (.881)	86.5% (.859)	85.6% (.849)	81.4% (.805)	89.0% (.884)
Subtopic *N* = 226	81.0% (.800)	78.3% (.771)	73.6% (.722)	71.9% (.705)	81.0% (.800)

Note. Results are based on using approximately 187,000 human-labeled cases to train the classifier to predict approximately 187,000 other cases (that were also labeled by humans but not used for training). Agreement is computed by comparing the machine's prediction to the human assigned labels. (AC1 measure presented in parentheses).

AC1 is intercoder reliability corrected for chance agreement.

Improvement over SVM alone is real,

Results of Hillard et al.

TABLE 3. Bill Title Interannotator Agreement for Five Model Types

	SVM	MaxEnt	Boostexter	Naïve Bayes	Ensemble
Major topic *N* = 20	88.7% (.881)	86.5% (.859)	85.6% (.849)	81.4% (.805)	89.0% (.884)
Subtopic *N* = 226	81.0% (.800)	78.3% (.771)	73.6% (.722)	71.9% (.705)	81.0% (.800)

Note. Results are based on using approximately 187,000 human-labeled cases to train the classifier to predict approximately 187,000 other cases (that were also labeled by humans but not used for training). Agreement is computed by comparing the machine's prediction to the human assigned labels. (AC1 measure presented in parentheses).

AC1 is intercoder reliability corrected for chance agreement.

Improvement over SVM alone is real, though small.

Results of Hillard et al.

TABLE 3. Bill Title Interannotator Agreement for Five Model Types

	SVM	MaxEnt	Boostexter	Naïve Bayes	Ensemble
Major topic *N* = 20	88.7% (.881)	86.5% (.859)	85.6% (.849)	81.4% (.805)	89.0% (.884)
Subtopic *N* = 226	81.0% (.800)	78.3% (.771)	73.6% (.722)	71.9% (.705)	81.0% (.800)

Note. Results are based on using approximately 187,000 human-labeled cases to train the classifier to predict approximately 187,000 other cases (that were also labeled by humans but not used for training). Agreement is computed by comparing the machine's prediction to the human assigned labels. (AC1 measure presented in parentheses).

AC1 is intercoder reliability corrected for chance agreement.

Improvement over SVM alone is real, though small.

Classification especially good in cases where methods agree on topic.

Results of Hillard et al.

TABLE 3. Bill Title Interannotator Agreement for Five Model Types

	SVM	MaxEnt	Boostexter	Naïve Bayes	Ensemble
Major topic N = 20	88.7% (.881)	86.5% (.859)	85.6% (.849)	81.4% (.805)	89.0% (.884)
Subtopic N = 226	81.0% (.800)	78.3% (.771)	73.6% (.722)	71.9% (.705)	81.0% (.800)

Note. Results are based on using approximately 187,000 human-labeled cases to train the classifier to predict approximately 187,000 other cases (that were also labeled by humans but not used for training). Agreement is computed by comparing the machine's prediction to the human assigned labels. (AC1 measure presented in parentheses).

AC1 is intercoder reliability corrected for chance agreement.

Improvement over SVM alone is real, though small.

Classification especially good in cases where methods agree on topic.

Exercise

Exercise

1. In real (deep) learning problems, do we want the learners in the ensemble to be similar or diverse relative to each other (in terms of architectures, hyper-parameters etc)? Why?

2. Ensembles give us better (more accurate) predictions, but they also give us more stable predictions. Why?

Overview of PCA

Overview of PCA

Features: these principal components will be uncorrelated (orthogonal) with (to) each other, Features: these principal components will be uncorrelated (orthogonal) with (to) each other, will be linear combinations of original variables

Overview of PCA

Features: these principal components will be uncorrelated (orthogonal) with (to) each other, will be linear combinations of original variables

Result: lower dimensional 'map' of observations in new space:

Overview of PCA

Features: these principal components will be uncorrelated (orthogonal) with (to) each other, will be linear combinations of original variables

Result: lower dimensional 'map' of observations in new space:

$\rightarrow$ each observation now has a value on each principal component called its (factor) score,

Overview of PCA

Features: these principal components will be uncorrelated (orthogonal) with (to) each other, will be linear combinations of original variables

Result: lower dimensional 'map' of observations in new space:

$\rightarrow$ each observation now has a value on each principal component called its (factor) score, which are projections of (original) observations onto the PCs

Overview of PCA

Features: these principal components will be uncorrelated (orthogonal) with (to) each other, will be linear combinations of original variables

Result: lower dimensional 'map' of observations in new space:

$\rightarrow$ each observation now has a value on each principal component called its (factor) score, which are projections of (original) observations onto the PCs

Interpretation of given PC: depends on correlation between component and (original) variable—known as loading

Overview of PCA

Features: these principal components will be uncorrelated (orthogonal) with (to) each other, will be linear combinations of original variables

Result: lower dimensional 'map' of observations in new space:

$\rightarrow$ each observation now has a value on each principal component called its (factor) score, which are projections of (original) observations onto the PCs

Interpretation of given PC: depends on correlation between component and (original) variable—known as loading

Method: (eigen-) decomposition of cov matrix or singular value decomposition of data matrix

Overview of PCA

Features: these principal components will be uncorrelated (orthogonal) with (to) each other, will be linear combinations of original variables

Result: lower dimensional 'map' of observations in new space:

$\rightarrow$ each observation now has a value on each principal component called its (factor) score, which are projections of (original) observations onto the PCs

Interpretation of given PC: depends on correlation between component and (original) variable—known as loading

Method: (eigen-) decomposition of cov matrix or singular value decomposition of data matrix

Exercise

Exercise

Consider the following quiz:

`http://psychcentral.com/quizzes/narcissistic.htm`

googling `psychcentral npi` seems to get there.

Think about how you would respond to the questions,

Exercise

Consider the following quiz:

`http://psychcentral.com/quizzes/narcissistic.htm`

googling `psychcentral npi` seems to get there.

Think about how you would respond to the questions, and fill them in privately if you wish!

Narcissistic Personality Inventory Items and Principal-Component Loadings

Items	Loadings 1	2	3	4	5	6	7
47. I would prefer to be a leader.	.83	.00	−.07	.04	−.12	.07	.22
15. I see myself as a good leader.	.83	.16	.09	−.12	.06	.03	−.14
13. I will be a success.	.67	.00	−.09	−.14	−.14	.17	.26
46. People always seem to recognize my authority.	.66	.02	.06	−.06	.06	.00	.20
2. I have a natural talent for influencing people.	.66	−.15	.02	−.02	.29	.03	−.24
16. I am assertive.	.56	.18	−.02	.22	−.02	−.03	−.27
17. I like to have authority over other people.	.56	.08	−.08	.18	.08	.05	.24
50. I am a born leader.	.35	.20	.22	.00	.09	−.14	−.01
30. I rarely depend on anyone else to get things done.	.02	.61	−.17	.04	.04	.10	−.11
23. I like to take responsibility for making decisions.	.28	.59	−.23	.23	−.12	.00	.02
53. I am more capable than other people.	−.19	.57	.16	.07	.11	.01	.20
45. I can live my life in any way I want to.	−.13	.46	.29	−.02	.05	.05	−.03
29. I always know what I am doing.	.15	.46	−.14	−.03	.30	.01	−.09
48. I am going to be a great person.	.05	.43	.39	.04	−.03	−.05	.00
54. I am an extraordinary person.	.06	.22	.69	−.07	−.06	.01	.06
7. I know that I am good because everybody keeps telling me so.	−.18	.01	.69	.00	.21	.01	.15
36. I like to be complimented.	.00	−.28	.67	.06	.00	.11	−.17
14. I think I am a special person.	.08	.16	.64	−.02	−.09	.17	−.01
51. I wish somebody would someday write my biography.	−.06	−.01	.57	.06	−.22	.09	.00
28. I am apt to show off if I get the chance.	−.04	−.02	.04	.71	−.03	.06	.06
3. Modesty doesn't become me.	−.01	.19	−.01	.69	−.16	−.06	.14
52. I get upset when people don't notice how I look when I go out in public.	−.16	.04	.10	.51	.09	.25	.17

Exercise

Exercise

1 We use principal components when we think there is a latent dimension(s) .

Exercise

1 We use principal components when we think there is a latent dimension(s) . What is an example of a dimension of interest in your work?

Exercise

1. We use principal components when we think there is a latent dimension(s) . What is an example of a dimension of interest in your work?

2. PCA is doing two things, intuitively: finding variable combinations that differ most across observations, and finding the combinations which predict the original data the best.

Exercise

1 We use principal components when we think there is a latent dimension(s) . What is an example of a dimension of interest in your work?

2 PCA is doing two things, intuitively: finding variable combinations that differ most across observations, and finding the combinations which predict the original data the best. These are equivalent: why?