1. Theoretical ML:

a. Apakah yang dimaksud dengan overfitting? Apa akibatnya?

Overfitting: Training accuracy > test eccuracy, terlalu mengikuti error/noise terlalu dekat

Hasil yang didapat tidak menghasilkan estimasi yang akurat dari respon untuk observasi baru yang bukan bagian dari training dataset

b. Mengapa kita fokus ke test error, bukan pada training error?

Karen test error yang digunakan untuk melakukan prediksi – how well it predict

c. Bagaimana kita mensimulasikan test error? (cross validation)

Cross-validation can be used to estimate the test error associated with a given statistical learning method in order to evaluate its performance, or to select the appropriate level of flexibility.

2. Bias-Variance Tradeoff:

a. Jelaskan dekomposisi dari training-error! (bias-variance)

The **bias–variance decomposition** is a way of analyzing a learning algorithm's [expected](https://en.wikipedia.org/wiki/Expected_value) [generalization error](https://en.wikipedia.org/wiki/Generalization_error) with respect to a particular problem as a sum of three terms, the bias, variance, and a quantity called the *irreducible error*, resulting from noise in the problem itself.

*E*(*y*0 *−* ˆ *f*(*x*0))2= Var( ˆ *f*(*x*0)) + [Bias( ˆ *f*(*x*0))]2 +Var(*e*)*.*

b. Bagaimana hubungan bias training error dengan model complexity?

Model complexity meningkat maka bias turun, vice versa

c. Bagaimana hubungan variance training error dengan model complexity?

Model complexity meningkat maka variance naik, vice versa

d. Bagaimana hubungan test-error dengan model comlexity?

Terdapat titik inklinasi, awalnya ketika kompleksitas bertambah error turun, namun kemudian makin kompleks modelnya akan makin naik

e. Bagaimana mencari model complexity paling tepat untuk mendapat test-error terendah?

Menggunakan cross validation

f. Bagaimana hubungan antara jumlah observasi dengan test error?

Semakin kecil jumlah observasi, nilai test error akan makin sulit dipercaya

Semakin banyak semakin baik

3. KNN:

a. Jelaskan cara KNN melakukan klasifikasi step-by-step!

- itung jarak dari X ke obs lain

- urutin jarak dari kecil ke besar

-pilih k

-dari k yang dipilih, itung majority vote (modus)

b. Jelaskan cara KNN melakukan regresi step-by-step!

- itung jarak dari X ke obs lain

- urutin jarak dari kecil ke besar

-pilih k

-dari k yang dipilih dirata2

c. Mengapa KNN dapat terkena Curse of Dimensionality?

given observation has no *nearby neighbors*—this is the so-called *curse of dimensionality*. That is, *K* observations that are nearest to a given test observation *x*0 may be very far away from *x*0 in *p*-dimensional space when *p* is large, leading to a very poor prediction of *f*(*x*0) and hence a poor KNN fit.

Intinya ketika Pnya besar (high dimension) hal ini akan membuat prediksinya makin jelek

d. Bagaimanakah cara meningkatkan model complexity pada KNN?

Menurunkan k, misal dari k = 100 menjadi k = 1

4. Linear Regression:

a. Bagaimana cara Linear Regression dalam melakukan prediksi?

(build a linear function from input to output which minimize training error)

Membuat fungsi input dan output yang meminimisasi training error

b. Bagaimana cara Linear Regression untuk melakukan optimisasi?

(closed form and SGD)

Dengan menggunakan closed form dan SGD

SGD : cari learning rate dan random parameter untuk beta, tentuin epoch (iteration), update sampe converge

c. Bagaimana cara intepretasi coefficient model Linear Regression: Y = 4 + 3X1?

4 = intercept – merupakan nilai Y jika X1 = 0

3X1 = slope – setiap kenaikan pada X1 maka Y akan mengalami kenaikan sebesar 3

d. Bagaimana cara menentukan variable selection yang meminimalisir error pada Linear Regression?

(Backward Selection, Forward Selection, Hybrid Selection)

BS = We start with all variables in the model, and remove the variable with the largest p-value—that is, the variable that is the least statistically significant. (buat model – buang)

FW = one adds variables to the model one at a time. At each step, each variable that is not already in the model is tested for inclusion in the model. The most significant of these variables is added to the model, so long as it's P-value is below some pre-set level. (masukin semua dari awal)

Mixed = buat, kalo ga cocok bisa buang

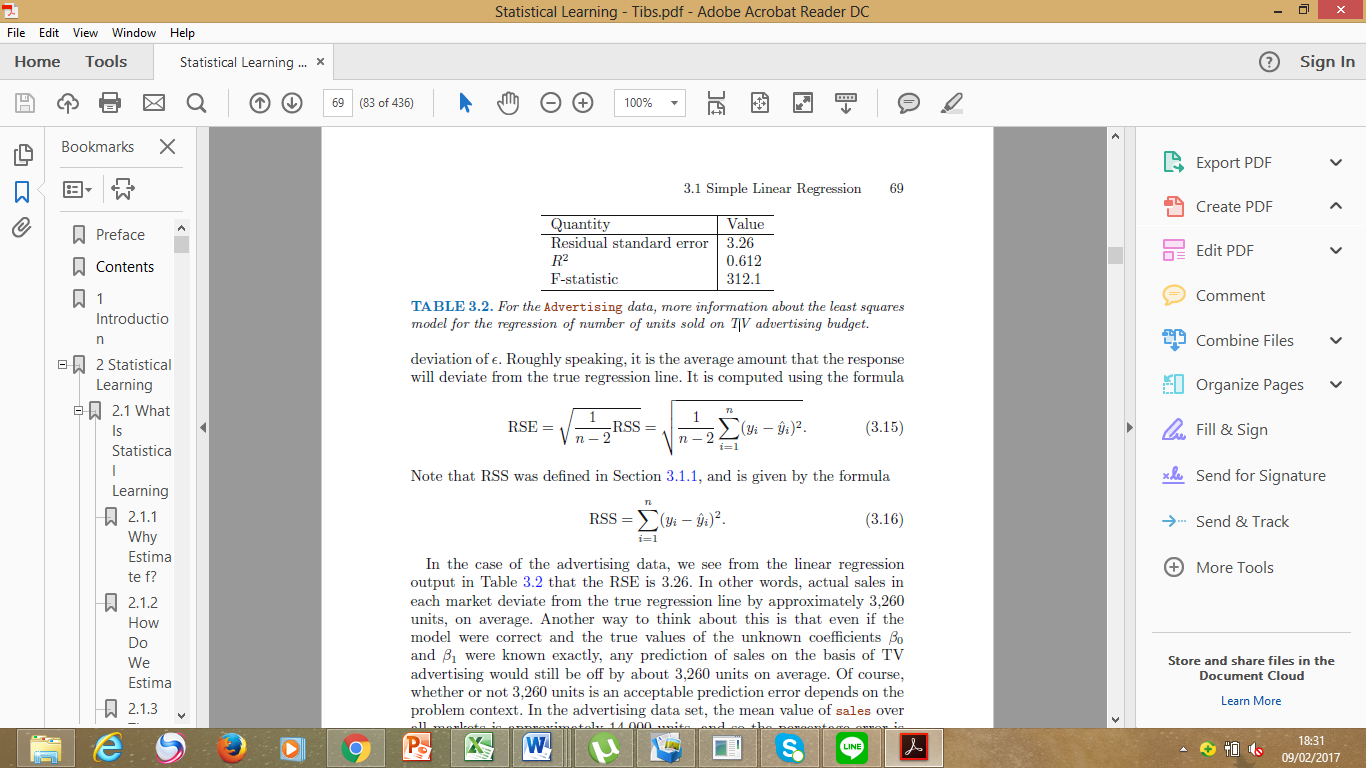
e. Bagaimana cara meningkatkan model complexity dari Linear Regression?

(tambahkan polynomial feature/input)

f. Bagaimana mencari model complexity paling tepat untuk mendapat test-error terendah pada Linear Regression?

Subset selection, cross validation

g. Apakah quantifikasi error dari Linear Regression?



5. Ridge Regression:

a. Apa perbedaan dari Ridge dan Lasso Regression dengan Linear Regression?

Ridge & Lasso with regularizer

Linear Regression without regularizer

b. Apa fungsi utama dari Ridge dan Lasso Regression?

Mengurangi overfit, pada Lasso dengan variable selection

c. Apa perbedaan koefisien hasil dari Ridge dan Lasso?

L1 – Lasso koefisien hasil mungkin nol

L2 – Ridge koefesien hasil hampir nol

d. Bagaimana cara menentukan hyperparameter dari L1 dan L2 yang pas untuk Lasso dan Ridge?

Menggunakan cross validation

6. Logistic Regression:

a. Apa perbedaan Logistic Regression dengan Linear Regression?

Logistic untuk klasifikasi, dimana rumus dari linear regression dimasukkan kedalam rumus sigmoid

b. Bagaimana cara Logistic Regression melakukan classification?

Rumus dari linear regression dimasukkan kedalam rumus sigmoid

c. what is the main functionality of sigmoid?

Dengan memberi batas 0 -1 terhadap rumus linear regression

d. what is the error of logistic regression?

(negative log likelihood loss)

e. Bagaimana cara meningkatkan complexity dari Logistic Regression?

Menambahkan feature dan menggunakan L1 & L2 norm

f. Apakah Logistic Regression dapat menggunakan L2 norm regulerizer?

Bisa, karena rumus log regression berasal dari linear regression

7. SVM:

a. Bagaimana cara Linear SVC dalam melakukan klasifikasi?

Menggunakan hyperplane dan maximal margin classifier

b. Bagaimana cara Linear SVC dalam melakukan klasifikasi jika data tidak linearly separable?

Menggunakan slack dan kernel

c. Bagaimanakah cara Kernel bekerja?

Dengan cara menambahkan dimensinya (polynomial) jadi bisa buat dimensi baru

d. Bagaimanakah cara meningkatkan model complexity dari SVM Kernel?

Menggunakan feature dan menaikan slack

8. Decision Tree:

a. Bagaimana cara Decision Tree melakukan klasifikasi?

Sama kaya regresi(lihat jawaban b) tapi dia menggunakan majority vote

b. Bagaimana cara Decision Tree melakukan regresi?

1. Use recursive binary splitting to grow a large tree on the training

data, stopping only when each terminal node has fewer than some

minimum number of observations.

2. Apply cost complexity pruning to the large tree in order to obtain a

sequence of best subtrees, as a function of *α*.

3. Use K-fold cross-validation to choose *α*. That is, divide the training

observations into *K* folds. For each *k* = 1*, . . .,K*:

(a) Repeat Steps 1 and 2 on all but the *k*th fold of the training data.

(b) Evaluate the mean squared prediction error on the data in the

left-out *k*th fold, as a function of *α*.

Average the results for each value of *α*, and pick *α* to minimize the

average error.

4. Return the subtree from Step 2 that corresponds to the chosen value

of *α*.

c. Apakah yang dimaksud dengan Weak Learning?

Memiliki hasil 0,5 + Ƞ

0 < Ƞ < 0,5

d. Mengapa Decision Tree hampir selalu overfitting?

Karena buat tree terus sampe untuk hal terkecil

9. Bagging dan Random Forest:

a. Bagaimana cara menurunkan overfitting dari Decision Tree?

Menggunakan bagging

b. Mengapa Bagging dapat menurunkan variance?

Karena variance dibagi dengan jumlah (n) yang semakin bertambah karena bootstrap

c. Apakah jumlah tree atau bagging berpengaruh pada overfitting?

Tidak

d. Apakah perbedaan dari Bagging dan Random Forest?

bagging = bootsrap semua terus averaging

RF= Melalukan dekorelasi (random variable)

e. Mengapa Random Forest melakukan random variable estimation?

Supaya trees nya tidak similar karena kalau similar akan lead to highly corelated tress dan highly corelated trees tidak akan berpengaruh banyak pada turunnya varians

10. Boosting:

a. Apakah perbedaan dari Bagging dan Random Forest dan Boosting?

Bagging = Bootsrap yang kemudian dilakuka aggregating

RF = dekorelasi (pilih random variable)

boosting = sequentially memperbaiki model

b. Apakah yang dimaksud dengan Strong Learning?

Memiliki hasil 1 – δ

0 < δ < 0,5

c. Bagaimana boosting bekerja?

Membangun model secara sequential dan mem-bypass error nya

d. Apakah Boosting dapat overfitting?

Iya, karena terus menerus mengoreksi error dengan penyesuaian weight dari model

SELAMAT BELAJAR