A1

Name: Axel Rooth
Persnr: 980602-3157

a) 3)
b) 9)
c) 6)
d) 10)
e) 8)

A2.

a) 16% on test because there is no error for training 1NN.

b) $\frac{10+x}{2} = 12 \Rightarrow x = 14\%$

c) $16\% + 12\% > 14\% + 12\%$
$\Rightarrow$ Use AdaBoost with 8% cv-error compared to 1NN 9% cv-error.

A3. Name: Axel Rooth
Pers nr: 980602-3157

a) Lasso Regression

b) ii) λ help to minimize overfitting which in other words minimize variance. Therefore, when $\lambda \to 0$ the variance steadily increase.

c) The property which makes Lasso a sparse model is that make penalization as irrelevent w_i become 0 which makes the model sparse.

A4.

a) The projection length $\sqrt{s}$ where
$$S = \sum_{i=1}^{P} (\bar{x}, \bar{u}_i)^2$$

b) The projection length shall always be maximized and therefor should x belong to class α.

Name: Axel Rosth
Persnr: 980602-3157

A5.

a) $-\log_2\left(\frac{13}{52}\right) = \underline{\underline{2}}$ bits

b) $E = -\sum p_i \log_2(p_i) - \{P(W) = \frac{2}{24}, P(L) = 1 - P(W)\}$

$= -\frac{2}{16}\log_2\left(\frac{2}{16}\right) - \frac{14}{16}\log_2\left(\frac{14}{16}\right)$

$\approx \underline{\underline{0{,}544 \text{ bits}}}$

c) $G_1 = E_P - \left(\frac{1}{2}\cdot 0 - \frac{1}{2}\log\left(\frac{1}{2}\right)\right) \approx \underline{\underline{0{,}0436}}$ bits

Answer: a) 2 bits b) 0,544 bits c) 0,0436 bits

B1. Name: Axel Rooth Persnr: 980602-3157

$P(\neg C) = x \Rightarrow P(C) = 1-x$, C = Covid

, $\neg C$ = Not Covid

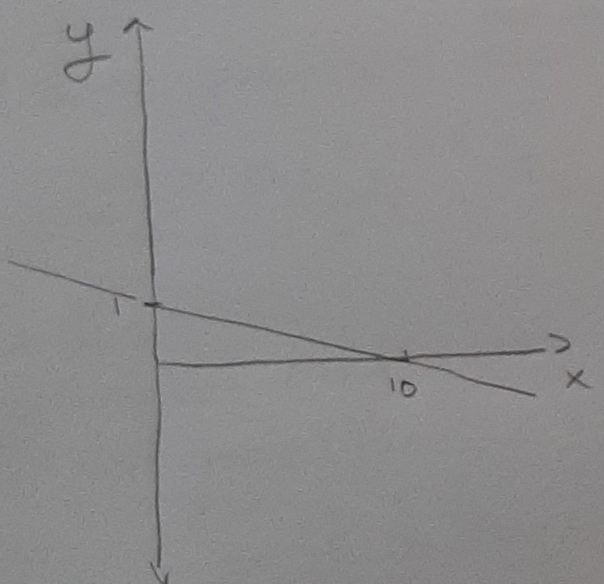
$P(+|\neg C) = 0{,}1$

$P(C|+) = y$

$$P(+|\neg C) = \frac{P(\neg C|+)}{P(\neg C)} = \frac{1-P(C|+)}{P(\neg C)} \quad (3)$$

$$0{,}1 = \frac{1-y}{x} \Rightarrow 0{,}1x = 1-y$$

$$\Rightarrow y = \underline{\underline{-0{,}1x+1}}$$



B2.

Name: Axel Rooth Persnr: 980602-3157

$$P(X=x) = \frac{\lambda_k^x e^{-\lambda_k}}{x!} \quad , x>0$$

$$L(\lambda_k) = \prod P(x=x)$$

$$l(\lambda_k) = \ln(L(\lambda_k)) = \sum \ln(\lambda_k^{x_i}) + -\lambda_k - \ln(x!)$$

$$= -n\lambda_k - \sum \ln(x!) + \ln(\lambda_k)\sum x_i$$

$$\frac{\partial l(\lambda_k)}{\partial \lambda_k} = -n + \frac{1}{\lambda_k}\sum x_i = 0$$

$$\Rightarrow \frac{1}{\lambda_k}\sum x_i = n \Rightarrow \lambda_k = \frac{1}{n}\sum x_i$$

Shoogee Nulnul; $\lambda = 0,5$
Max the tax; $\lambda = 2,5$

$$P(\text{same poopoos}) = \sum P(SN = p_i) \cdot P(MTT = p_i)$$

$$= \sum \frac{0,5^x e^{-0,5}}{x!} \cdot \frac{2,5^x e^{-2,5}}{x!} \approx \underline{\underline{0,134}}$$

Code:

```
import math
def map():
    p=0
    for i in range(100):
        p += 0.5**i * math.exp(-0,5)/math.factorial(i)*\
        2,5**i * math.exp(-2,5)/math.facturia(i)
    return p
```

Answer: SN $\lambda = 0,5$ MTT $\lambda = 2,5$
P(same poo) $\approx 0,134$

B3

$$P(\lambda_k | \text{data}) = \frac{P(\text{data} | \lambda_k) P(\lambda_k)}{P(\text{data})}$$

$$p(\lambda_k | \text{data}) = \ln P(\lambda_k | \text{data})$$

$$= \ln P(\text{data} | \lambda_k) + \ln P(\lambda_k) - \ln P(\text{data})$$

$$= l(\lambda_k) + \ln \lambda_k e^{-\gamma \lambda_k} - \ln P(\text{data})$$

$$= -n\lambda_k - \sum \ln(x!) + \ln(\lambda_k) \sum x_i$$
$$+ \ln \lambda_k - \gamma \lambda_k - \ln P(\text{data})$$

$$\frac{\partial p(\lambda_k | \text{data})}{\partial \lambda_k} = -n + \frac{1}{\lambda_k} \sum x_i + \frac{1}{\lambda_k} - \gamma = 0$$

$$\Longrightarrow \gamma = \frac{1}{\lambda_k}(1 + \sum x_i) - n$$

Answer:

$\Longrightarrow$ Shoogee $\gamma = 3.5$

Max the Tax $\gamma = 1.5$

Name: Axel Rooth

Persnr: 980602-3157

B4. Name: Axel Rooth Persnr: 980602-3157

$$P(y|x,w) = \text{Poisson}(wx+b)$$

$$P(X=x) = \frac{\lambda_k^{x} e^{-\lambda_k}}{x!}$$

$$P(y|x,w) =$$

C1

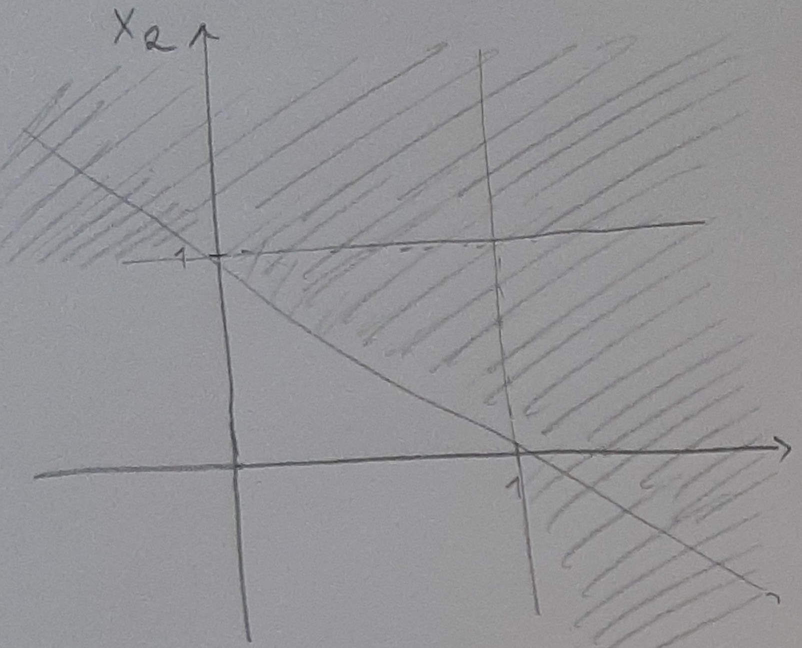
a) A, C, F, H, G

b) In favor: With a non-linear kernal, a non-linear decission bounday can be made. This can therefore result in a better classification because of a larger margin.

Against: Because of the non-linear kernal, one can expect to use more computing power of result of a more complex model.

C2.

Name: Axel Rooth Persnr: 980602-3157



a) Positive in grey
Negative in white □

b) No, it cannot. The reason is of the non-linearity of the classification output space.

c) Nothing changes if all the weight doubles.

$x_1 = 1 \longrightarrow 4 \; f_1(\cdot) = \{4-4=0\} = 1$

4 -4

$x_2 = 0$

b

You can also multiply with 3, 4, ...
The output stays the same for $\hat{w}_i = \lambda w_i$ where λ is a scalar.