1 Estimating Parameters

1.1 Closed Form Estimation

1. Likelihood function is

$$L(\lambda; x) = P(D \mid \lambda) = \prod_{i=1}^{n} P(X_i \mid \lambda)$$
$$= \lambda^n e^{-\lambda \sum_i x_i}.$$

The derivative of likelihood function is

$$\frac{\partial L(\lambda; x)}{\partial \lambda} = \lambda^{n-1} e - \lambda \sum_{i=1}^{n} x_i (n - \lambda \sum_{i=1}^{n} x_i).$$

Because $x_i > 0$, by setting derivative to zero, we know that the maximum likelihood estimate of λ is

$$\hat{\lambda}_{MLE} = \arg\max_{\lambda} L(\lambda; x) = \frac{n}{\sum_{i=1}^{n} x_i}.$$

To prove biasness of $\hat{\lambda}_{MLE}$, we need to calculate $\mathbf{E}(\lambda) = \mathbf{E}(n/\sum_i X_i)$

Now let $Y = \sum_{i=1}^{n} X_i$. Because X_i follows exponential distribution, Y follows gamma distribution with shape parameter a equals to n and rate parameter b equals to $1/\lambda$. That is to say,

$$p(y) = \frac{\lambda^n}{\Gamma(n)} y^{n-1} e^{-\lambda y}.$$

Therefore

$$\mathbf{E}(1/\sum_i X_i) = \mathbf{E}(1/Y) = \int_0^{\infty} \frac{\lambda^n y^{n-1}}{y\Gamma(n)} e^{-\lambda y} dy = \int_0^{\infty} \frac{\lambda^n y^{n-2}}{\Gamma(n)} e^{-\lambda y} dy$$
$$= \frac{\lambda}{n-1} \int_0^{infty} \frac{(\lambda y)^{n-2}}{\Gamma(n-1)} e^{-(\lambda y)} d(\lambda y)$$
$$= \frac{\lambda}{n-1}$$

.

Thus

$$\mathbf{E}(\hat{\lambda}) = \mathbf{E}(n/\sum_i X_i) = \mathbf{E}(n/Y) = \frac{n}{n-1}\lambda$$

i.e., $\hat{\lambda}_{MLE}$ is biased.

2. Let i.i.d data $\{x_i\}_{i=1}^n \sim Exp(\lambda)$, then

$$p(\lambda \mid x_1, ..., x_n) \propto p(x_1, ..., x_n \mid \lambda) p(\lambda)$$
$$\propto \prod_i \lambda e^{-\lambda x_i} \frac{\beta^\alpha}{\Gamma(\alpha)} \lambda^{\alpha-1} e^{-\beta\lambda}$$
$$\propto \frac{\beta^\alpha}{\Gamma(\alpha)} \lambda^{n+\alpha-1} e^{-(\beta + \sum_i x_i)\lambda}$$
$$\propto \lambda^{n+\alpha-1} e^{-(\beta+\sum_i x_i)\lambda}.$$

So $P(\lambda \mid x) \sim Gamma(n+\alpha-1, \beta+\sum_i x_i)$ follows a gamma distribution.
Now let $L(\lambda, x) = \lambda^{n+\alpha-1} e^{-(\beta+\sum_i x_i)\lambda}$, then

$$\frac{\partial L(\lambda, x)}{\partial \lambda} = \lambda^{n+\alpha-2} e^{-(\beta+\sum_i x_i)}[n+\alpha-1-(\beta+\sum_{i=1}^{n} x_i)\lambda].$$

By setting above function to 0, we have

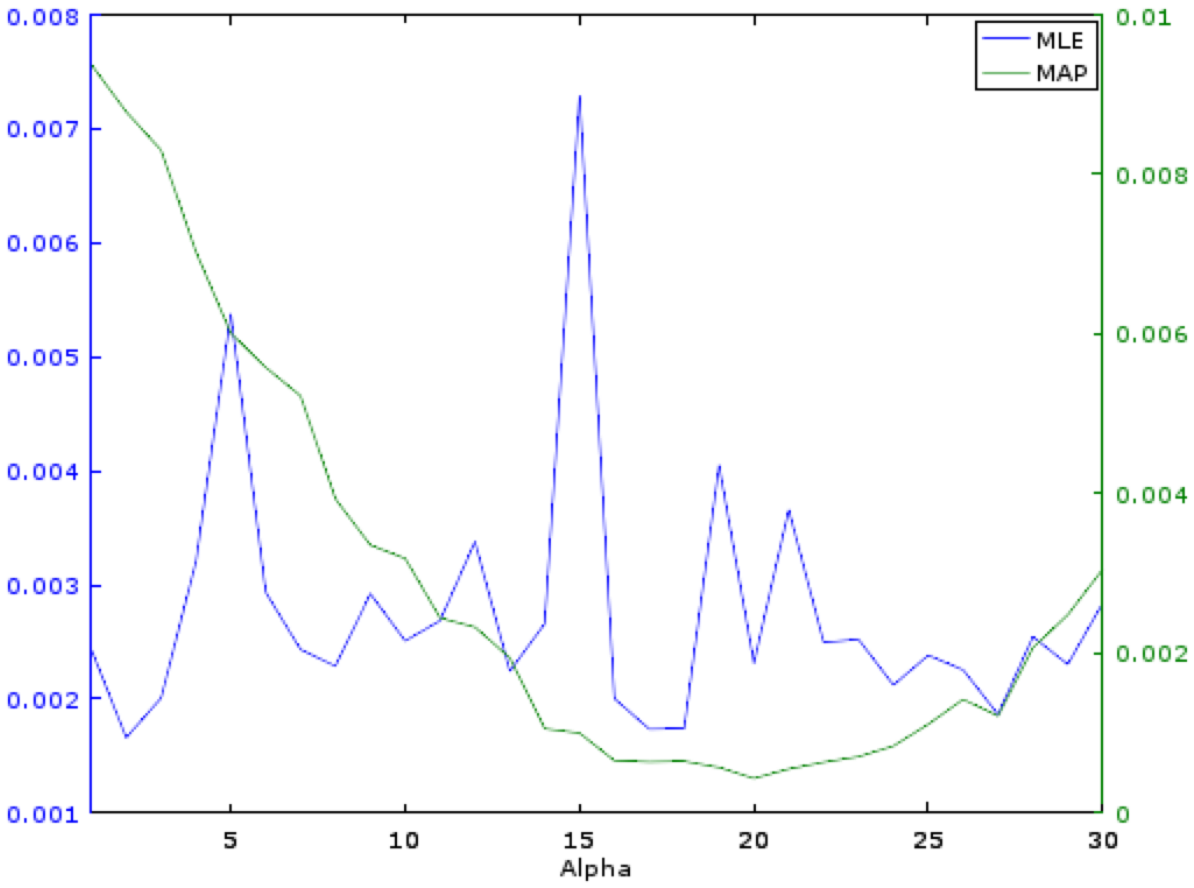
$$\hat{\lambda}_{MAP} = \frac{1+(\alpha-1)/n}{\beta/n+\sum x_i/n}.$$

Thus

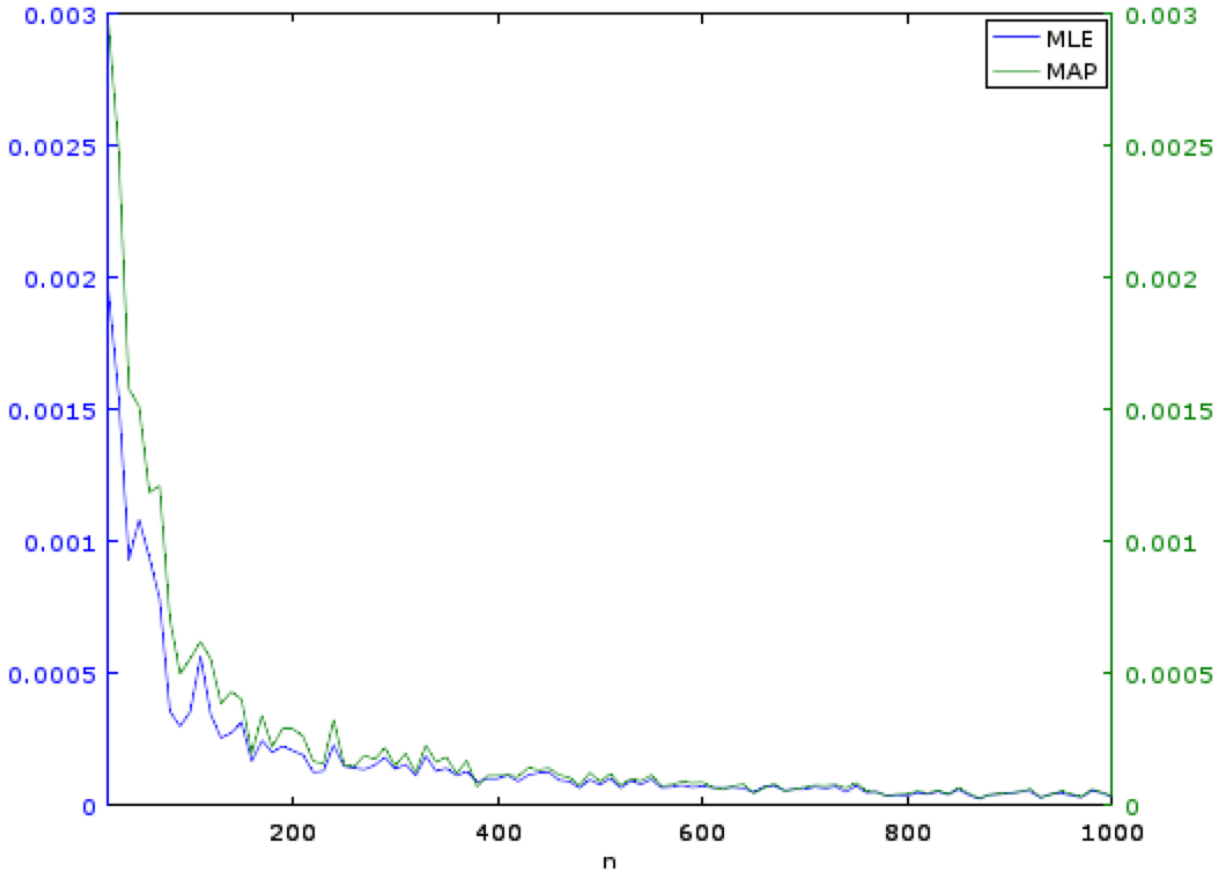
$$\lim_{n\to\infty} \hat{\lambda}_{MAP} = 1/\bar{x} = 1/\mu = \lambda,$$

i.e., $\hat{\lambda}_{MAP}$ converges to true λ.

3.



4.



5. (a) When prior distribution is accurate, MAP estimator is better. As can be seen from plot in question 3, mean squared error for MAP estimator is significantly lower. This is because at this point, the expected value of $\hat{\lambda}_{MAP}$ happens to be $\alpha/\beta = 20/100 = 0.2$, which is exactly the true λ.

   (b) On the other hand, if the given prior distribution is not as accurate as above, and the number of samples n is small, then as can be seen from plot in question 4, MLE is generally better than MAP.

1.2 Non-closed Form Estimation

1. Because i.i.d. data $\{x_i\}_{i=1}^n \sim Gamma(\alpha, \beta)$, the likelihood function is

$$\begin{aligned} L(\alpha, \beta; x_1, ..., x_n) &= \prod_{i=1}^{n} p(x_i \mid \alpha, \beta) \\ &= \prod_{i=1}^{n} [\frac{\beta^\alpha}{\Gamma(\alpha)}]^n (\prod_{i=1}^{n} x_i)^{\alpha-1} e^{-\beta \sum_i x_i}. \end{aligned}$$

   The log-likelihood function is therefore

$$ln(L) = n[\alpha ln\beta - \psi(\alpha)] + (\alpha - 1) \sum_{i=1}^{n} lnx_i - \beta \sum_{i=1}^{n} x_i,$$

where $\psi(\alpha)$ is the digamma function. Now take the derivative log-likelihood function,

$$\frac{\partial lnL}{\partial \alpha} = n[ln\beta - \psi(\alpha)] + \sum_{i=1}^{n} lnx_i,$$
$$\frac{\partial lnL}{\partial \beta} = n\frac{\alpha}{\beta} - \sum_{i=1}^{n} x_i.$$

Thus at each step, we update α and β by the following rule

$$\alpha \leftarrow \alpha + \gamma\frac{\partial lnL}{\partial \alpha} = \alpha + \gamma(n(ln\beta - \psi(\alpha)) + \sum_{i=1}^{n} lnx_i),$$
$$\beta \leftarrow \beta + \gamma\frac{\partial lnL}{\partial \beta} = \beta + \gamma(n\frac{\alpha}{\beta} - \sum_{i=1}^{n} x_i).$$

The final result from calculation is

$$\hat{\alpha}_{MLE} = 4.404213$$
$$\hat{\beta}_{MLE} = 0.505892$$

2. Let
$$F_1 = \frac{\partial lnL}{\partial \alpha}, F_2 = \frac{\partial lnL}{\partial \beta}.$$

Then it's easy to calculate the Jacobian matrix

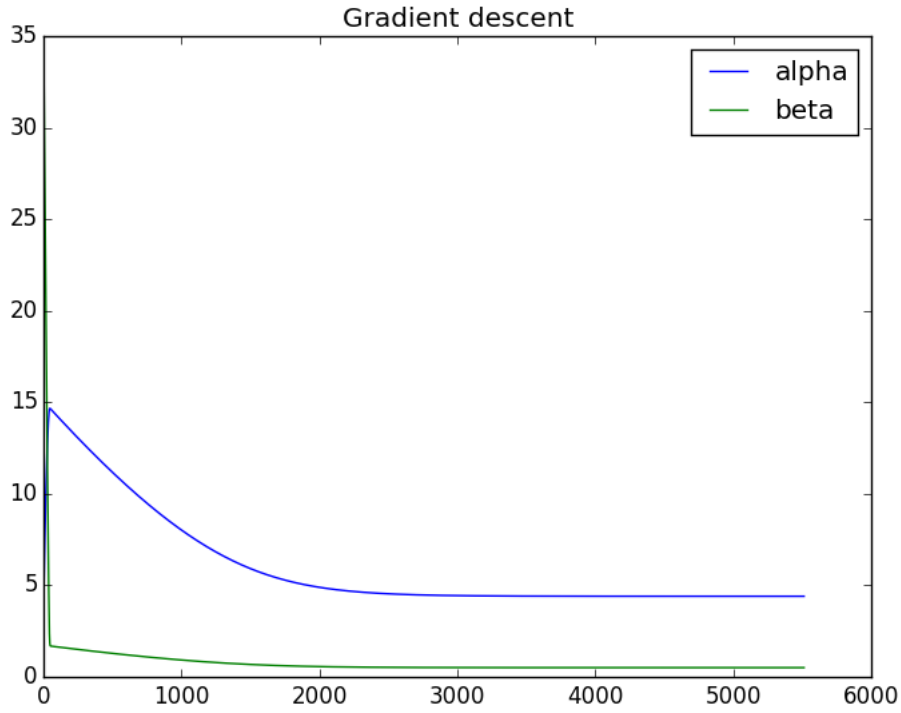
$$J_F(\alpha, \beta) = \begin{pmatrix} -n\psi'(\alpha) & n/\beta \\ n/\beta & -n\alpha/\beta^2 \end{pmatrix}$$

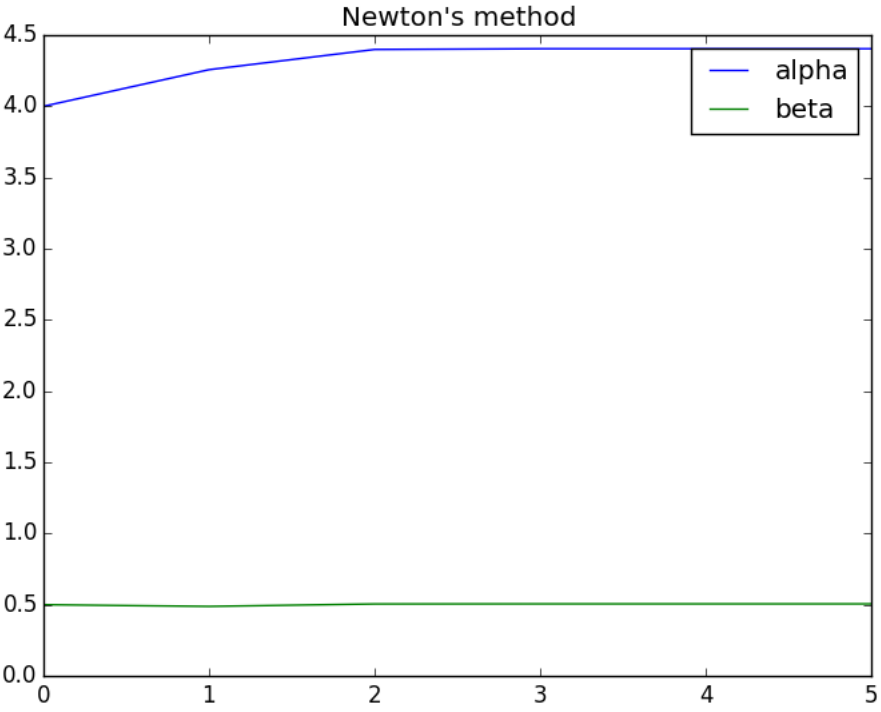
From the previous question we know that the true value of α and β should be somewhere around 4 and 0.5. Therefore, we set the initial value for Newton's method to 4 and 0.5.

Below are results from Newton's method:

Iteration	α	β
1	4.0	0.5
2	4.257106	0.4880238
3	4.399032	0.5052629
4	4.404170	0.5058874
5	4.404176	0.5058881
6	4.404176	0.5058881

3. Implementation and results from gradient descent and Newton'e method can be seen from above two questions. The convergence plots are as follows:





Newton's method apparently converges much faster, this is mainly because Newton's method uses information from second-order derivatives. But this is also because we need to set the initial value of Newton close to the true root (otherwise Newton's method

encounters invalid value problems).

The actual estimated values are $\alpha = 4.404$ and $\beta = 0.505$.

2 Naive Bayes

2.1 Simple Naive Bayes

See code.

2.2 Bayes Model Averaging

1. Likelihood function L is

$$\begin{aligned}\sum_f P(x, f|y, D)P(y|D) &= \sum_f P(x|f, y, D)P(f|y, D)P(y|D) \\ &= \sum_f P(x|f, y, D)P(f|D)P(y|D).\end{aligned} \tag{1}$$

   The last step is because y and f are independent.

2. According to Bayes rule, we have

$$\begin{aligned}P(f|D) = \frac{P(D|f)P(f)}{P(D)} \propto P(D|f)P(f) &= P(f)\prod_{i=1}^{N} P(y^{(i)}, x^{(i)}|f) \\ &= P(f)\prod_{i=1}^{N} P(y^{(i)})P(x^{(i)}|f, y^{(i)}).\end{aligned}$$

   Plug this into equation (1), and relabel the new observation (x, y) as $(x^{(N+1)}, y^{(N+1)})$, we then have

$$\sum_f P(x|f, y)P(y)P(f|D) = \sum_f P(f)\prod_{i=1}^{N+1} P(y^{(i)})P(x^{(i)}|f, y^{(i)}).$$

3. For a Naive Bayes Classifier, label y is chosen such that

$$P(y|x, D) \propto P(y, x, D) = P(y)\prod_{k=1} KP(x_k|y)$$

   is maximized.

On the other hand,

$$\begin{aligned}
P(y|x,D) &\propto \sum_f P(f) \prod_{i=1}^{N+1} P(y^{(i)}) \prod_{k=1}^{K} P(x_k^{(i)}|f_k, y^{(i)}) \\
&= \sum_f \left[\prod_{k=1}^{K} P(f_k) \right] \prod_{i=1}^{N+1} P(y^{(i)}) \prod_{k=1}^{K} P(x_k^{(i)}|f_k, y^{(i)}) \\
&= \sum_{f_1} \sum_{f_2} \cdots \sum_{f_K} \left[\prod_{i=1}^{N+1} P(y^{(i)}) \right] \prod_{k=1}^{K} P(f_k) \prod_{k=1}^{K} P(x_k^{(i)}|f_k, y^{(i)}) \\
&= \left[\prod_{i=1}^{N+1} P(y^{(i)}) \right] \prod_{k=1}^{K} \sum_{f_k} P(f_k) \prod_{i=1}^{N+1} P(x_k^{(i)}|f_k, y^{(i)}) \\
&= \left[\prod_{i=1}^{N+1} P(y^{(i)}) \right] \prod_{k=1}^{K} \left[\prod_{i=1}^{N+1} P(x_k^{(i)}) + \frac{1}{\beta} \prod_{i=1}^{N+1} P(x_k^{(i)}|y^{(i)}) \right]
\end{aligned}$$

4. See code.

Refrence

Wu, Ga, Sanner, Scott, AND Oliveira, Rodrigo. "Bayesian Model Averaging Naive Bayes (BMA-NB): Averaging over an Exponential Number of Feature Models in Linear Time" AAAI Conference on Artificial Intelligence (2015)

3 Convexity

3.1 Basic definition

(a)

1. *Proof.* $\Rightarrow$: It's obvious, just let t be $1/2$.
   $\Leftarrow$: Suppose f is not a convex function. Then there exist $a, b \in \mathbb{R}$ and $t_0 \in (0,1)$ such that

$$f(t_0 a + (1-t_0)b) > t_0 f(a) + (1-t_0) f(b).$$

Let $g(x) : [a,b] \to \mathbb{R}$ be

$$g(x) = f(x) - \frac{f(b)-f(a)}{b-a}(x-a) - f(a).$$

Now that

$$\begin{aligned}
g(t_0 a + (1-t_0)b) &= f(t_0 a + (1-t_0)b) - \frac{f(b)-f(a)}{b-a}(t_0 a + (1-t_0)b - a) - f(a) \\
&> t_0 f(a) + (1-t_0) f(b) - \frac{f(b)-f(a)}{b-a}(t_0 a + (1-t_0)b - a) - f(a) \\
&= t_0 f(a) + (1-t_0) f(b) - (1-t_0)[f(b) - f(a)] - f(a) \\
&= 0.
\end{aligned}$$

We therefore have $M = \max_{x\in[a,b]} g(x) > 0$. What's more, it's easy to notice that $g(a) = g(b) = 0$, and

$$\begin{aligned} g(\frac{x+y}{2}) &= f(\frac{x+y}{2}) - \frac{f(b)-f(a)}{b-a}(\frac{x+y}{2} - a) - f(a) \\ &\leq \frac{f(x)+f(y)}{2} - \frac{f(b)-f(a)}{b-a}(\frac{x+y}{2} - 2\frac{a}{2}) - 2\frac{f(a)}{2} \\ &= \frac{g(x)+g(y)}{2}. \end{aligned}$$

Let $c \in [a, b]$ be the smallest value such that $g(c) = M$, and δ be a small enough value such that $c \pm \delta \in (a, b)$. Thus we have

$$g(c) = g(\frac{c-\delta+c+\delta}{2}) \leq \frac{g(c-\delta)+g(c+\delta)}{2} < \frac{g(c)+g(c)}{2} = g(c),$$

which is a contradiction. □

2. *Proof.* $\Rightarrow$: Be the definition of derivative, we have

$$\begin{aligned} f'(x)(y-x) + f(x) &= \lim_{t\to 1^+} \frac{f(tx+(1-t)y) - f(x)}{(1-t)(y-x)}(y-x) + f(x) \\ &= \lim_{t\to 1^+} \frac{f(tx+(1-t)y) - f(x)}{1-t} + f(x) \\ &\leq \lim_{t\to 1^+} \frac{tf(x) - (1-t)f(y) - f(x)}{1-t} + f(x) \\ &= \lim_{t\to 1^+} f(y) \\ &= f(y). \end{aligned}$$

$\Leftarrow$: Let $z = tx + (1-t)y$, then we have

$$\begin{aligned} f(x) &\geq f(z) + f'(z)(x-z) \\ f(y) &\geq f(z) + f'(z)(y-z) \end{aligned}$$

Multiply them with t and $1-t$ and add the two equations, we have

$$\begin{aligned} tf(x) + (1-t)f(y) &\geq f(z) + f'(z)(tx + (1-t)y - z) \\ &= f(z) + f'(z)(z-z) \\ &= f(tx + (1-t)y). \end{aligned}$$

□

3. *Proof.* $\Rightarrow$: Suppose f is convex, and $x < y$, then by the previous question, we have

$$\begin{aligned} f(y) &\geq f(x) + f'(x)(y-x) \\ f(x) &\geq f(y) + f'(y)(x-y). \end{aligned}$$

Because $x < y$, reformat the above equations and we will have

$$\begin{aligned} f'(x) &\leq \frac{f(y)-f(x)}{y-x}, \\ f'(y) &\geq \frac{f(y)-f(x)}{y-x}, \end{aligned}$$

i.e. $f'(x) \leq f'(y)$. Therefore, $f'(x)$ is non-decreasing, that is equivalent to $f''(x) \geq 0, \forall x \in \mathbb{R}$.

$\Leftarrow$: We use Taylor's series expansion of f around some point x_0:

$$f(x) = f(x_0) + f'(x_0)(x - x_0) + \frac{f''(x^*)}{2}(x - x0)^2,$$

where x^* lies between x_0 and x. Since $f''(x^*) \geq 0$, the last term is nonnegative for all x.
Let $x_0 = tx + (1-t)y$, then

$$f(x) \geq f(x_0) + f'(x_0)[(1-t)(x-y)].$$

Now let $x = y$, we obtain

$$f(y) \geq f(x_0) + f'(x_0)[t(y-x)].$$

Multiply the two equations by t and $1-t$ and add them together, we will have

$$tf(x) + (1-t)f(y) \geq f(x_0) = f(tx + (1-t)y).$$

Thus we proved f is convex. □

(b)

1. *Proof.* Since

$$\begin{aligned} h(tx + (1-t)y) &= \max\{f(tx + (1-t)y), g(tx + (1-t)y)\} \\ &\leq \max\{tf(x) + (1-t)f(y), tg(x) + (1-t)g(y)\} \\ &= t\max\{f(x), g(x)\} + (1-t)\max\{f(y), g(y)\} \\ &= th(x) + (1-t)h(t), \end{aligned}$$

it's easy to notice this proved that $h(x)$ is a convex function. □

2. *Proof.*

$$\begin{aligned} h(tx + (1-t)y) &= f(tx + (1-t)y) + g(tx + (1-t)y) \\ &\leq tf(x) + (1-t)f(y) + tg(x) + (1-t)g(y) \\ &= t(f(x) + g(x)) + (1-t)(f(y) + g(y)) \\ &= th(x) + (1-t)h(y). \end{aligned}$$

□

3. f(g(x)) is not necessarily a convex function. A counterexample is $x^{2/3} = (-x^{1/3})^2$. It's concave on $(0, \infty)$, but both x^2 and $-x^{1/3}$ are convex on $(0, \infty)$.
To make $f(g(x))$ convex, we must have f, g convex and f nondecreasing.
Suppose f and g satisfy the above conditions, then by convexity of g, we have

$$g(tx + (1-t)y) \leq tg(x) + (1-t)g(y).$$

In addition, since f is nondecreasing,

$$f(g(tx + (1-t)y) \leq f(tg(x) + (1-t)g(y)).$$

Again, by convexity of f, we have

$$f(tg(x) + (1-t)g(y)) \leq tf(g(x)) + (1-t)f(g(x)).$$

Therefore, $f(g(x))$ is a convex function.
An intuitive explanation can be seen from the case when both f and g are twice differentiable.

3.2 Functions used in deep learning

1. Yes. Because the summation and max of convex functions are still convex, we only need to prove that $1 - y_i r_i$ is convex. This is obvious because it is a linear function.
2. Yes because linear function is convex and the max of two convex functions is still a convex function.
3. No. $1 + e^{-r_i}$ is convex but $-log(x)$ is neither convex nor nondecreasing. Therefore, the composition is not convex.
4. Yes. First of all, e^x is convex and nondecreasing, so $e^{r_i(j)}$ is convex. Since the summation of convex functions is still convex, $\sum_{j=1}^{k} e^{r_i(j)}$ is convex. In addition, $log(x)$ is convex and nondecreasing on $(0, \infty)$, thus $log(\sum_{j=1}^{k} e^{r_i(j)})$ is convex. While $r_i(s)$ is just a linear term, adding it won't change the convexity. Therefore, $S_s(\mathbf{W})$ is a convex function.
5. Because neural networks often use non-convex activation function such as log-sigmoid and tangent sigmoid. These non-convex activation functions will result to non-convex problems.