1. (a) Up & more overfitting: deterministic noise depends on the target function f, so if the complexity of f increases, the deterministic noise so generally increase as well. There is less overfitting when the target complexity is low, in this case it is increasing, so there's a higher tendency to overfit the data.

   (b) Up & less overfitting: deterministic noise will generally go up because, relative to the fixed target function, H becomes less complex. Target complexity is exponential when compared to overfitting, whereas noise is linear, so there will be less overfitting.

2. (a) We try to satisfy the condition at 4.4, which tells us that $w^T w \leq C$, to convert $w^T \Gamma^T \Gamma w \leq C$ to $w^T w \leq C$, **Γ has to be the identity vector**, then the inverse of Γ and its dot product is 1. Anytime we have a scalar 1, we are left with the an unchanged matrix. We are left with $w^T w$, which is equivalent to $\sum_{q=0}^{Q} w_q^2 \leq C$.

   (b) We can consider **Γ to be a row vector with values 1**, this will create the following matrix:

$$\begin{bmatrix} w_1 \\ w_2 \\ . \\ . \\ . \\ w_n \end{bmatrix} \begin{bmatrix} 1 & 1 & . & . & . & 1 \end{bmatrix} \times \begin{bmatrix} w_1 & w_2 & . & . & . & w_n \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ . \\ . \\ . \\ 1 \end{bmatrix}$$

   We can solve this matrix to get

$$(w_1 + w_2 + ... + w_n) \times (w_1 + w_2 + ... + w_n) = (w_1 + w_2 + ... + w_n)^2$$

   From this, we know that we can represent $(w_1 + w_2 + ... + w_n)^2$ in terms of summations with $(\sum_{q=0}^{Q} w_q)^2$, so if $w^T \Gamma^T \Gamma w \leq C$ and we chose our Γ to be the row vector with values 1, then it is also the case that $(\sum_{q=0}^{Q} w_q)^2 \leq C$.

3. Hard-order constraint: we have the perceptron model which uses a linear model, we can define this as the hypothesis set H_2. When we compare H_2 and a higher order hypothesis set, let that be H_{10}, on a dataset with a lot of noise and low N, H_2 will have a smaller out of sample error due to its tendency to not overfit compared to H_{10}. So we place a hard-order constraint constraint on it and choose the simpler hypothesis, leading to a smaller d_{vc} and a smaller E_{out}.

4. (a)

$$\begin{aligned}
\sigma_{val}^2 &= var_{d_{val}}[E_{val}(g^-)] \\
&= var_{d_{val}}[\frac{1}{K}\sum_{x_n \in D_{val}} e(g^-(x), y)] \\
&= \frac{1}{K} var_{d_{val}}[\sum_{x_n \in D_{val}} e(g^-(x), y)] \\
&= \frac{1}{K} var_{d_{val}}[e(g^-(x_n), y)] \\
&= \frac{1}{K} var_{x_n}[e(g^-(x_n), y)]
\end{aligned}$$

For x, we have:

$$\frac{1}{K} var_x[e(g^-(x), y)]$$

We know that $\sigma^2(g^-) = var_x[e(g^-(x), y)]$, so $\sigma_{val}^2 = \frac{1}{k}\sigma^2(g^-)$.

(b)

$$\begin{aligned}
\sigma_{val}^2 &= \frac{1}{K} var_x[e(g^-(x), y)] \\
&= \frac{1}{K} var_x[g(x) \neq y]
\end{aligned}$$

We know $var_x = p(1-p)$, where $p = P[g(x) \neq y]$, so we can derive the following:

$$\begin{aligned}
var_x[g(x) \neq y] &= (P[g(x) \neq y]) \times (1 - P[g(x) \neq y]) \\
\sigma_{val}^2 &= \frac{1}{K}(P[g(x) \neq y]) \times (1 - P[g(x) \neq y]) \\
&= \frac{1}{K}(P[g(x) \neq y] - P[g(x) \neq y]^2)
\end{aligned}$$

(c) We maximize our probability to $\frac{1}{2}$ to give us the highest variance.

$$\begin{aligned}
\sigma_{val}^2 &= \frac{1}{K}(p - p^2) \\
&= \frac{\frac{1}{2} - (\frac{1}{2})^2}{K} \\
&= \frac{\frac{1}{4}}{K} \\
&= \frac{1}{4K}
\end{aligned}$$

Since we have maximized the value p, then we have found the upper bound such to prove $\sigma_{val}^2 \leq \frac{1}{4K}$

(d) **No**, since there is no bound for the squared error, lets assume an arbitrary large error, meaning $g(x)$ and y are very far apart. Our Var depends on squared error, if squared error is unbounded, then Var will similarly be unbounded, making it so there is no upper bound.

(e) **Higher**, if you have fewer points, then you will have higher variance because the points can be more scattered. It is only when you have a large enough N that the outliers will be more meaningless, which established the average. With a lower number of points, we would have higher squared error, since variance depends on squared error, we should have a higher variance too.

(f) The more datapoints you use for validation, the more datapoints you lose out on creating your hypothesis g, which results in a larger E_{out}. The less datapoints you use for validation, the less expected error there would be since there will be a significant number of points used to calculate our hypothesis g, which results in a smaller E_{out}.

5. Yes, E_m does not affect the training set, so it is not biased in estimating the out of sample error E_{out}.