Q1

1.1.1: If batch size increase, then $E(g_B(w))$ remains unchanged, but $V(g_B(w)) = \frac{B}{B^2} V(g(w))$ will decrease.

Thus optimal learning rate is expected to INcrease as training becomes more stable.

1.1.2

a> point C; as at A, the benefit of increasing batch size is not saturated, while at point B, the benefit is over saturation ⇒ no significant decrease on training steps.

b> A: noise

B: ill-curvature

1.1.3

a> I, III-, IV

b> II+, III-

1.2:

a>(1) model A has more parameter, since it takes longer time before any updates, and achieved a smaller loss.

(2) model B has more iterations, as it has less parameter; thus given same computation time, B must have more iterations than A.

b> given each training step requires same amount of time to compute (piazza 673), model A is preferred as it requires less updates. Also hardware resources can be utilized more efficiently by training large models on GPU.

Q2

2.1.1

expand: $(w_*^T \tilde{x} - \hat{w}^T \tilde{x})^2$

$= (w_*^T \tilde{x})^T (w_*^T \tilde{x}) - 2(w_*^T \tilde{x})^T (\hat{w}^T \tilde{x}) + (\hat{w}^T \tilde{x})^T (\hat{w}^T \tilde{x})$

$= \tilde{x}^T w_* w_*^T \tilde{x} - 2\tilde{x}^T w_* \hat{w}^T \tilde{x} + \tilde{x}^T \hat{w} \hat{w}^T \tilde{x}$

(plug in $\hat{w} = (X^T X)^{-1} X^T t$, and $t = Xw_* + \varepsilon$) $\frac{(X^T X)^{-1} X^T X w_*}{(X^T X)^{-1} X^T \varepsilon}$

$= \tilde{x}^T w_* w_*^T \tilde{x} - 2\tilde{x}^T w_* w_*^T \tilde{x} - 2\tilde{x}^T w_* \varepsilon^T X((X^T X)^{-1})^T \tilde{x}$

$+ \tilde{x}^T (w_* + (X^T X)^{-1} X^T \varepsilon)(w_*^T + \varepsilon^T X((X^T X)^{-1})^T)\tilde{x}$

$= -\cancel{\tilde{x}^T w_* w_*^T \tilde{x}} - \cancel{2\tilde{x}^T w_* \varepsilon^T X((X^T X)^{-1})^T \tilde{x}}$

$+ \cancel{\tilde{x}^T w_* w_*^T \tilde{x}} + \tilde{x}^T \cancel{(X^T X)^{-1} X^T \varepsilon w_*^T \tilde{x}} + \tilde{x}^T \cancel{w_* \varepsilon^T X((X^T X)^{-1})^T \tilde{x}}$

$+ \tilde{x}^T (X^T X)^{-1} X^T \varepsilon \varepsilon^T X((X^T X)^{-1})^T \tilde{x}$

$= \tilde{x}^T (X^T X)^{-1} X^T \varepsilon \varepsilon^T X((X^T X)^{-1})^T \tilde{x}$ ⇒ realizing it's a scalar thus the trace of this expression is itself

Now consider expectation:

Expectation $= E[\tilde{x}^T (X^T X)^{-1} X^T \varepsilon \varepsilon^T X((X^T X)^{-1})^T \tilde{x}]$

$= E[tr(\tilde{x}^T (X^T X)^{-1} X^T \varepsilon \varepsilon^T X((X^T X)^{-1})^T \tilde{x})]$

$= tr(E[\tilde{x}^T (X^T X)^{-1} X^T \varepsilon \varepsilon^T X((X^T X)^{-1})^T \tilde{x}])$

$= tr(E[\tilde{x}\tilde{x}^T]\, E[(X^T X)^{-1} X^T]\, E[\varepsilon \varepsilon^T]\, E[X((X^T X)^{-1})^T])$

$= \sigma^2\, tr(I_d\, E[(X^T X)^{-1} X^T]\, I_n\, E[X((X^T X)^{-1})^T])$

$= \sigma^2\, tr(E[I_d (X^T X)^{-1} X^T I_n X] \cdot E[((X^T X)^{-1})^T])$

$$= \sigma^2 E[\,tr(((X^TX)^{-1})^T)\,] = \sigma^2\, tr((X^TX)^{-1}) \qquad (tr(A) = tr(A^T))$$

2.2.1:

$n > d$: $\qquad m > p+1 \qquad G \in R^{m \times p} \qquad Tr((G^TG)^{-1})$

$$E[R(\hat{w})] = 0 + \frac{\sigma^2 d}{n-d-1}$$

draft: $X \in R^{n \times d}$ $\quad n \lfloor_d \quad X^T: d \lfloor_n$ $\Rightarrow$ $Tr((X^TX)^{-1})$ can simply plug in formula;

$G: m \lfloor_p \quad G^T: p \lfloor_m$

$n < d$:

$$E[R(\hat{w})] = \underbrace{\frac{d-n}{d}}_{\text{bias}} + \underbrace{\sigma^2 \frac{n}{d-n-1}}_{\text{variance}}$$

draft: $G: m \times p \quad m \lfloor_p \quad G^T: p \lfloor_m \qquad p = n$

$X: n \times d \quad n \lfloor_d \quad X^T: d \lfloor_n \qquad m = d$

$G^TG \Rightarrow XX^T$

2.2.2:

① conditions involve:

(1) $n < d$; $\quad$ (as when $n > d$, $\frac{d}{n-d-1}\sigma^2$ can never be zero unless $d = 0$ ...)

(2) $n - d = \sigma^2 \frac{n}{d-n-1}$ $\quad$ (derived directly from $n<d$'s $E[R(\hat{w})]$)

above is for $d > n$ case, but can lead to contradiction for (2)

$n - d < 0$, but $\sigma^2 \frac{n}{d-n-1} \geq 0 \Rightarrow$ impossible

now consider $n > d$;

then as long as $\sigma^2 = 0$, zero generalization loss could be achieved;

thus condition is: $n > d$ and $\sigma = 0$

② as more examples are added, "$|d-n-1|$" or "$|n-d-1|$" will become small enough so that when it divides 'n', the resulting generalization loss can be quite large, given σ^2 isn't changed.

2.3.2:

If sample size increases, λ should decrease, since more samples leads to more stable weights.

If noise level increases, λ should increase to penalize possible too large weights.

2.3.4:

a> with ridge-regularizer, the generalization loss consistently lowers while when without a regularizer, the loss dramatically increased a lot when # samples is close to # features.

b> with ridge-regularizer, adding more samples always lead to decrease of generalization error, and thus test error is also decreased.