1.1.1

Since as the batch size increases, the variance of the gradient noise would decrease, which means that the fluctuation caused by the stochasticity of the gradient descent. Therefore, the optimal learning rate would increase as the batch size increases since the fluctuation caused by the variations between mini-batches decrease.

1.1.2

a) Point C. Before point C, if we increase the batch size, the # of training steps to take decreases linearly. After point C, the increase of the batch size would bring relevant no decrease in # of training steps to take.

b)

Point A: Regime: noise dominated

Point B: Regime: curvature dominated

1.1.3

a) I, IV

b) II+, III-

1.2

(a)

(1) Model A. More parameters will lead to a finer model with smaller test loss, and as model A has smaller test loss than model B. model A has more parameters

(2) Since model A has more parameters. at the intersection "X", model B has been training for more iterations/updates, given the total compute calculation method

b) I will choose model A in any situation since with same total compute. as model B has less number of parameters, it would take more training steps than A, and so more wall-clock time than A. A can lead to a smaller test loss in the end with more time to train as well, but one can notice after "X", to reach the same test loss A would need much less total compute than B, leading to much less training steps.

2.1.1

Since $w_*^T\tilde{x} - \hat{w}^T\tilde{x}$ is a number, we can apply the trace operator to the expectation operator, and we have

$$E_{\tilde{x},\varepsilon,w_*}(w_*^T\tilde{x} - \hat{w}^T\tilde{x})^2 = E[\tilde{x}^T(w_* - \hat{w})(w_*^T - \hat{w}^T)\tilde{x}]$$

$$= E[tr(\tilde{x}^T(w_* - \hat{w})(w_*^T - \hat{w}^T)\tilde{x})]$$

$$= E[tr(\tilde{x}\tilde{x}^T(w_* - \hat{w})(w_*^T - \hat{w}^T))] \quad (\text{cyclic prop. of trace})$$

$$= tr[E(\tilde{x}\tilde{x}^T)E(w_*w_*^T - w_*\hat{w}^T - \hat{w}w_*^T + \hat{w}\hat{w}^T)] \quad (\tilde{x} \perp \hat{w} \& w_*)$$

$$= tr[E(\tilde{x}\tilde{x}^T)E(w_*w_*^T)] - tr[E(\tilde{x}\tilde{x}^T)E(w_*\hat{w}^T)] - tr[E(\tilde{x}\tilde{x}^T)E(\hat{w}w_*^T)] + tr[E(\tilde{x}\tilde{x}^T)E(\hat{w}\hat{w}^T)]$$

Now, since $\tilde{x} \sim N(0, I_d)$. we have $E(\tilde{x}\tilde{x}^T) = I_d$. By given, $E(w_* w_*^T) = \frac{1}{d} I_d$. With $\hat{w}$, since we have when $n > d$, $\hat{w} = (X^T X)^{-1} X^T t$, where $t = X w_* + \varepsilon$. Therefore, we have $\hat{w} = (X^T X)^{-1} X^T (X w_* + \varepsilon)$

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

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

Therefore, $E[w_* \hat{w}^T] = E(w_* [w_*^T + \varepsilon^T X (X^T X)^{-1}])$

$$= E(w_* w_*^T) + E[w_* \varepsilon^T X (X^T X)^{-1}]$$

$$= \frac{1}{d} I_d + 0 \quad (E(\varepsilon) = 0, \text{ and } w_*, \varepsilon, X \text{ are indep of each other})$$

Similarly, $E(\hat{w} w_*^T) = E[(w_* + (X^T X)^{-1} X^T \varepsilon) w_*^T]$

$$= E(w_* w_*^T) + E[(X^T X)^{-1} X^T \varepsilon w_*^T] = \frac{1}{d} I_d + 0$$

Now, $E(\hat{w}\hat{w}^T) = E[(w_* + (X^T X)^{-1} X^T \varepsilon)(w_*^T + \varepsilon^T X (X^T X)^{-1})]$

$$= E(w_* w_*^T + w_* \varepsilon^T X (X^T X)^{-1} + (X^T X)^{-1} X^T \varepsilon w_*^T + (X^T X)^{-1} X^T \varepsilon \varepsilon^T X (X^T X)^{-1})$$

$$= \frac{1}{d} I_d + E[(X^T X)^{-1} X^T \varepsilon \varepsilon^T X (X^T X)^{-1}]$$

So the four trace terms become:

$$1 - 1 - 1 + tr(I_d \frac{1}{d} I_d + I_d E[(X^T X)^{-1} X^T \varepsilon \varepsilon^T X (X^T X)^{-1}])$$

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

$$= 0 + E(I_d tr[(X^T X)^{-1} X^T X (X^T X)^{-1} \varepsilon \varepsilon^T)]$$

$$= 0 + tr(I_d E((X^T X)^{-1} \varepsilon \varepsilon^T)) = 0 + \sigma^2 Tr((X^T X)^{-1}) \quad (E(\varepsilon \varepsilon^T) = \sigma^2 I_n)$$

2.2.1

For underparametrized case, since $X \in \mathbb{R}^{n \times d}$ with $X_i \sim N(0, I_d)$. This means that $[X]_{ij} \sim N(0,1)$, and so we have

$$E[R(\hat{w})] = \sigma^2 E[Tr((X^T X)^{-1})]$$

$$= \sigma^2 \frac{d}{n-d-1}$$

For overparametrized case, we have:

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

Assume $A = X^T \in \mathbb{R}^{d \times n}$, we can use the second given property and getting:

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

2.2.2

(1) For underparametrized model, to set $E[R(\hat{w})] = 0$ one would get $d=0$ or $\sigma=0$, but $d=0$ means no parameters, which is not possible, and so for parametrized model, $E[R(\hat{w})] = 0$ when $\sigma=0$, i.e., it is noiseless. For overparametrized model, $E[R(\hat{w})] = 0$ either $n=d=0$ or $n=d$ & $\sigma=0$. But we have $d>n$, so in overparametrized model, $E[R(\hat{w})] \neq 0$. So

$$E[R(\hat{w})] = 0 \Leftrightarrow n \geq d \ \& \ \sigma = 0$$

(2) No. Even though adding more training examples would drive the model to underparametrized-model case where $E[R(\hat{w})] = 0$ has the solution, but one cannot make sure that the new training examples are noiseless, and more training examples typically mean more noise.

2.3.2

λ should increase as training set size n increases since as n increases, the model tends to overfitting with smaller variance, and so λ should

increase to prevent overfitting.

λ should decrease as noise level σ increases since the variance is already quite high due to large σ, and decrease λ would reduce the variance.

2.3.4

- The expectation of the trace now becomes the larger root to the quadratic equation: $f(x) = \lambda x^2 + (1-\gamma+\lambda) + \gamma = \lambda x^2 + (1-\frac{d}{n}+\lambda) + \frac{d}{n}$. Also, from the graph we see a huge jump when n is around $d=500$ without regularization, but the generalization error decreases as training set size n increases.
- Yes. From the graph we can see adding more training data always lead to better test performance.