Task 1

Completed in linear_regression.py.

Task 2

As λ approaches infinity the regularized component of the sum-of-squares error also goes to infinity. The only way to the error is minimized is if w gets smaller and smaller to offset λ. If w is anything non-zero, then the error is infinite so w must be zero.

The actual values of the sum-of-squares error equation does not matter because the regularized component of the equation completely offsets the actual error component. Regularization is usually focused on avoiding large values of w but as λ gets larger, the 'force' pushing w to be smaller gets more forceful. Eventually w approaches zero as seen on slides 63-64.

Task 3

- f(x) = 3.1x + 4.2
    - f(x1) = 26.03 vs 9.6
    - f(x2) = 31.61 vs 4.2
    - f(x3) = 29.44 vs 2.2
    - SE = .5*$((9.6-26.03)^2+(4.2-31.61)^2+(2.2-29.44)^2)$ = **886.11**
- f(x) = 2.4x - 1.5
    - f(x1) = 11.22 vs 9.6
    - f(x2) = 15.54 vs 4.2
    - f(x3) = 13.86 vs 2.2
    - SE = .5*$((9.6-11.22)^2+(4.2-15.54)^2+(2.2-13.86)^2)$ = **133.59**

The second function is better according to the sum-of-squares criterion because the sum of squared errors for the second function is less than the first function. 133.59 < 886.11.

Task 4

Bob's algorithm should not replace the current model because the optimal λ will always be zero in order to reduce the squared error value. Because Bob's algorithm will always train and set λ to zero, there is no regularization in his algorithm. While his weights will match the training set closer than the standard training algorithm, when testing against new data, Bob's algorithm has the possibility of being very off. Because λ is zero, some w values can be very large and overfit Bob's weights to the training set. The standard model will test better against newer data while Bob's model has the potential of having very large errors against new test data.