# Gradient Boosting Trees: The Math

$$z_i = -\frac{\partial Loss(y, F_i)}{\partial F_i}$$

**Negative Gradient of Loss w.r.t. Ensemble Prediction**

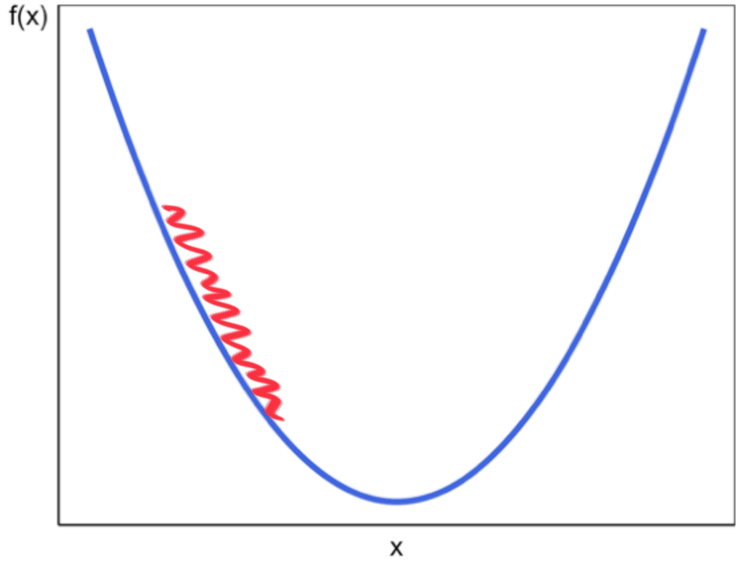
- The Negative Gradient tell us what adjustments we should make to our prediction $F_i$ in order to decrease our loss
- Example:

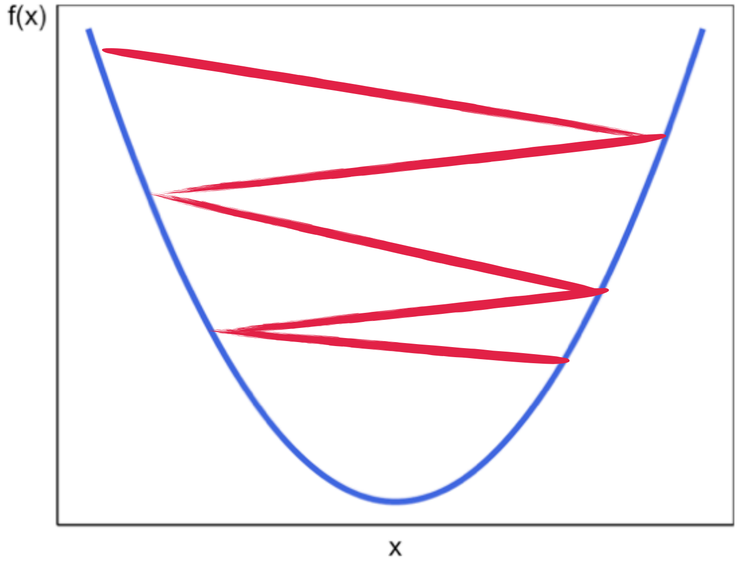
$$Loss(y, \hat{y}) = (y - \hat{y})^2 \implies -\frac{\partial Loss(y, \hat{y})}{\partial \hat{y}} \implies 2(y - \hat{y})$$

  - With squared loss, error is the negative gradient, but the negative gradient will work in other situations!

# Choosing a Learning Rate: Convexity

- Under ideal conditions, gradient descent iteratively approximates and converges to the optimum
- For a constant learning rate $\lambda$
  - if $\lambda$ is too small, it takes too many iterations to reach the optimum
  - if $\lambda$ is too large, algorithm may 'bounce' around the optimum and never get close




  - Better to treat learning rate as a variable, that is let the value depend on gradient
  - around optimum $\lambda$ is small, and far from optimum $\lambda$ is larger