Least square line fitting

Least squares fit to the red points
Least square line fitting: robustness to noise

Squared error heavily penalizes outliers
(RANdom SAmple Consensus): Learning technique to estimate parameters of a model by random sampling of observed data
Algorithm:
1. **Sample** (randomly) the number of points required to fit the model
2. **Solve** for model parameters using samples
3. **Score** by the fraction of inliers within a preset threshold of the model
4. **Reevaluate** the fit according to the inliers

Repeat 1-4 until the best model is found with high confidence Return the model with the most inliers
Algorithm:
1. **Sample** (randomly) the number of points required to fit the model
2. **Solve** for model parameters using samples
3. **Score** by the fraction of inliers within a preset threshold of the model
4. **Reevaluate** the fit according to the inliers

**Repeat** 1-4 until the best model is found with high confidence. **Return** the model with the most inliers.
Algorithm:
1. **Sample** (randomly) the number of points required to fit the model
2. **Solve** for model parameters using samples
3. **Score** by the fraction of inliers within a preset threshold of the model
4. **Reevaluate** the fit according to the inliers

**Repeat** 1-4 until the best model is found with high confidence. Return the model with the most inliers.
Algorithm:
1. **Sample** (randomly) the number of points required to fit the model
2. **Solve** for model parameters using samples
3. **Score** by the fraction of inliers within a preset threshold of the model
4. **Reevaluate** the fit according to the inliers

Repeat 1-4 until the best model is found with high confidence. Return the model with the most inliers.
Algorithm:
1. Sample (randomly) the number of points required to fit the model
2. Solve for model parameters using samples
3. Score by the fraction of inliers within a preset threshold of the model
4. Reevaluate the fit according to the inliers

Repeat 1-4 until the best model is found with high confidence Return the model with the most inliers
Algorithm:
1. **Sample** (randomly) the number of points required to fit the model
2. **Solve** for model parameters using samples
3. **Score** by the fraction of inliers within a preset threshold of the model
4. **Reevaluate** the fit according to the inliers

Repeat 1-4 until the best model is found with high confidence. Return the model with the most inliers.

$N_I = 14$