PlopQuiz
Assessment Problem

Given

- A set of quiz items, each quiz item having an unknown level of difficulty
- A set of quiz takers, each quiz taker having an unknown level of proficiency
- A set of scores, each score generated by a single quiz taker answering a single quiz item
- Set of scores not connecting all quiz items to all quiz takers

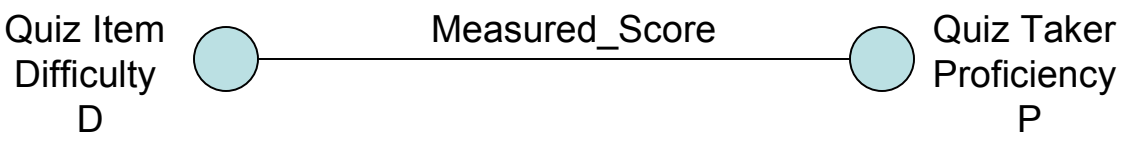
Problem

- Estimate the difficulties of each quiz item and the proficiency of each quiz taker.

Difference with conventional assessment problems

- No Standard Quiz is provided.
- Difficulties and Proficiencies must be calculated simultaneously

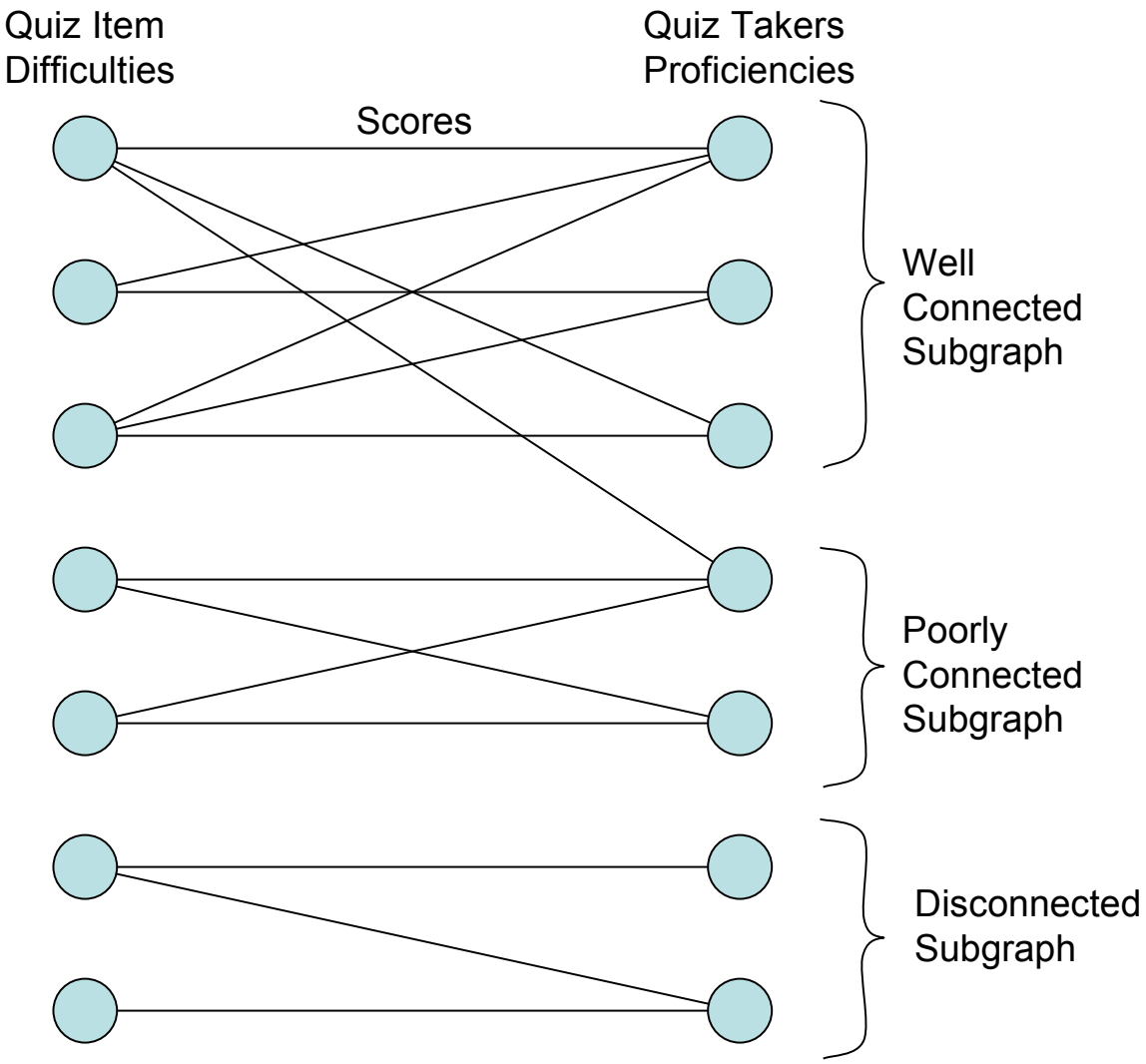
The Raw Data: Measured Scores



Each Measured Scores Relates
One Quiz Taker to One Quiz Item

Graph Representation

Assumption:
Quiz Items and Quiz Takers Not All Connected
Best if They Form a (Well) Connected Graph

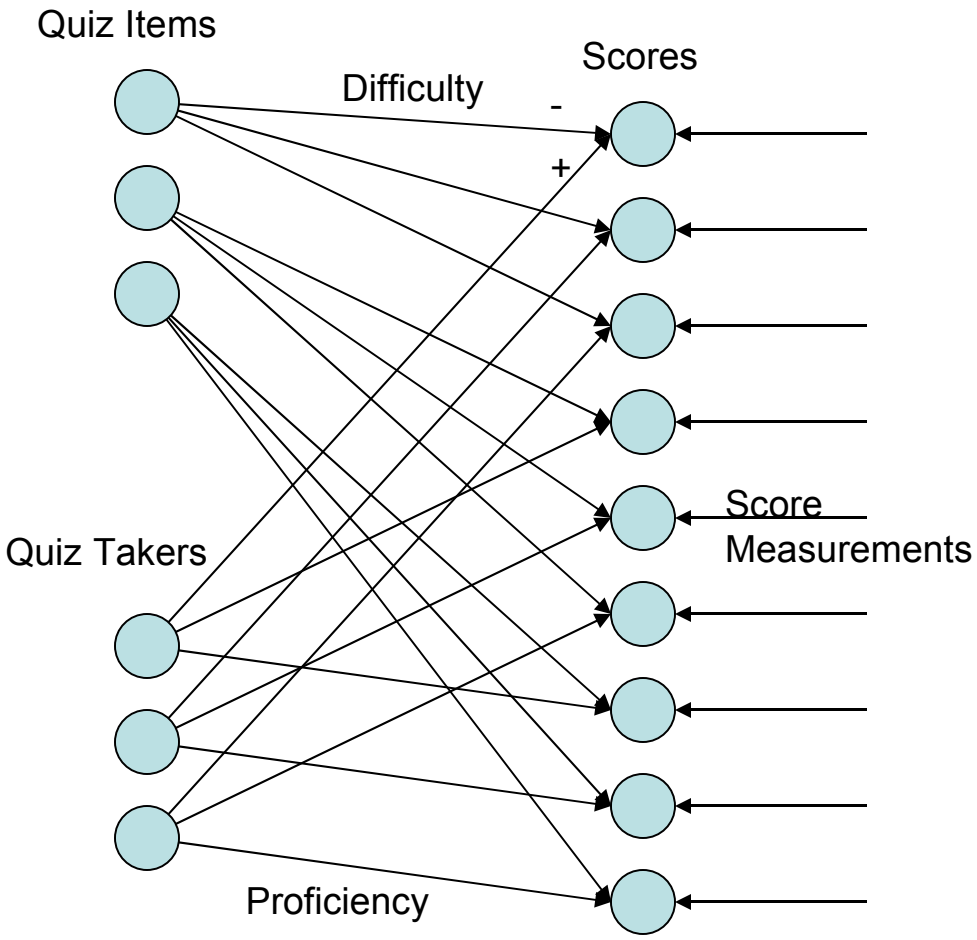


Possible Technical Approaches

1) Neural Net Backpropagation

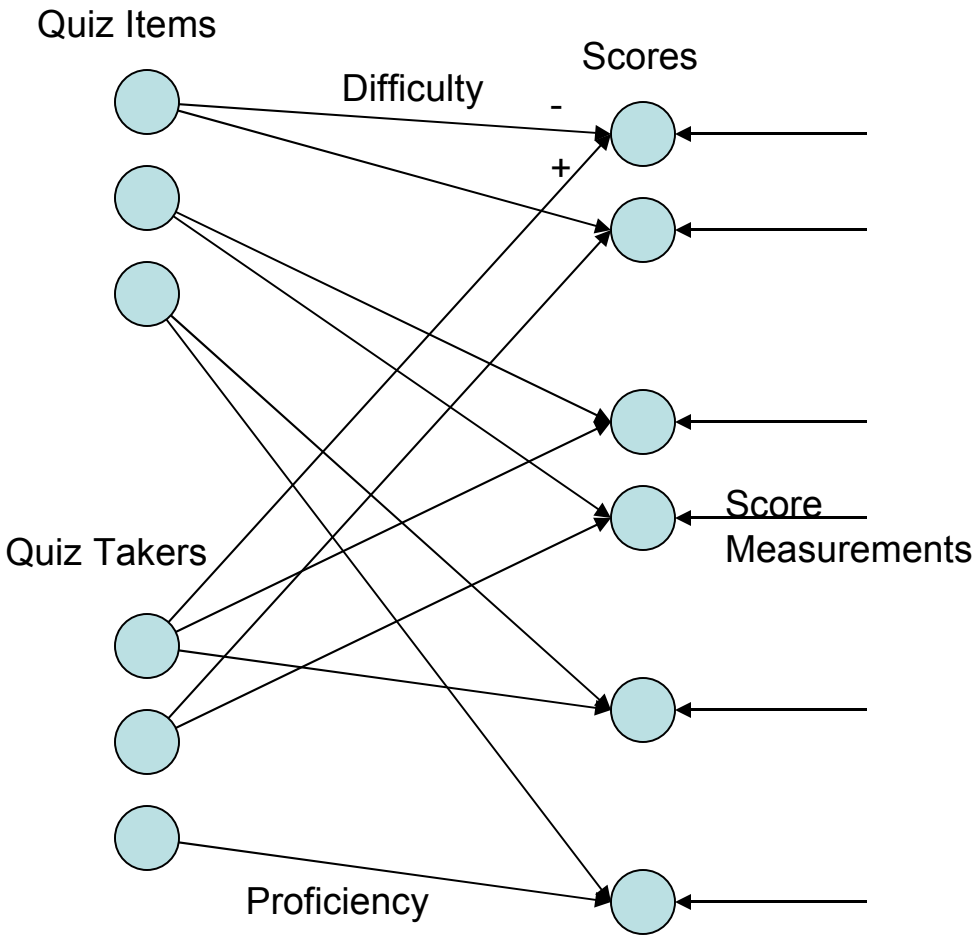
2) Kalman Filtering
(High speed variations include sub-optimal filtering using a predetermined computed covariance matrix)

Neural Net Representation (Fully Connected)



Backpropagation algorithm may be applicable

Neural Net Representation (Not Fully Connected)

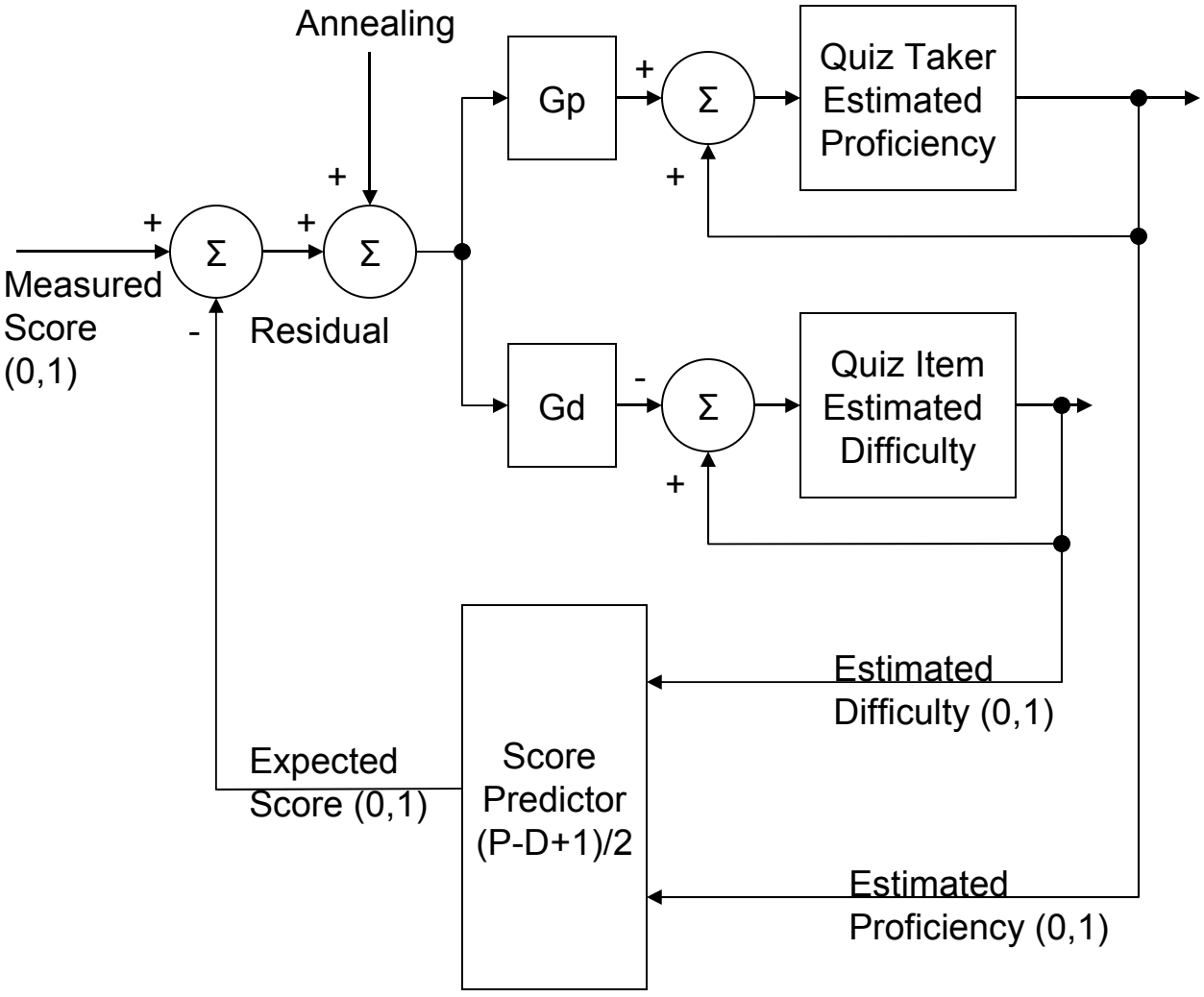


A Extended/Simpified Kalman Filter Can be Used

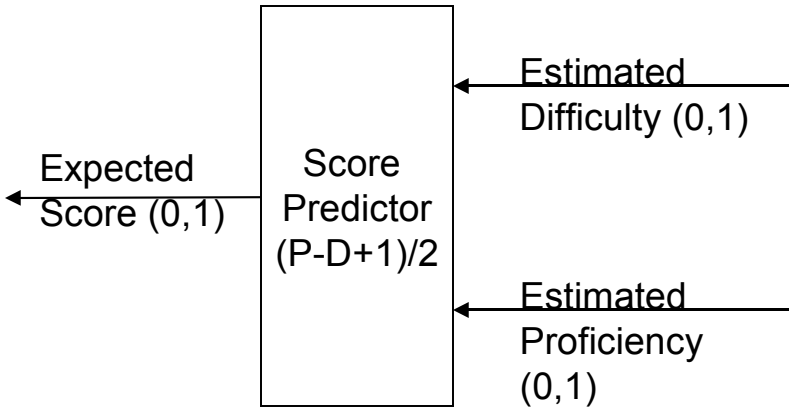
Output is:

Quiz Item Difficulties and

Quiz Takers Proficiencies



Expected Score Predictor Makes Use of Fuzzy Logic

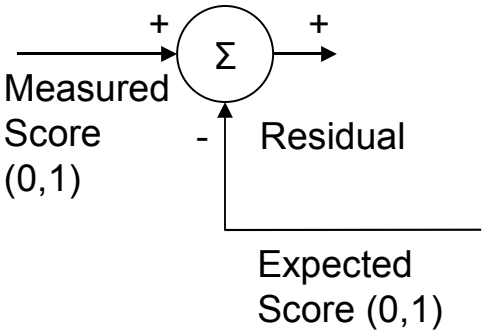


Pro-ficiency \ Difficul-ty	Low (0)	Medium (0.5)	High (1)
Low (0)	Medium (0.5)	Medium low (0.25)	Low (0)
Medium (0.5)	Medium High (0.75)	Medium (0.5)	Medium low (0.25)
High (1)	High (1)	Medium High (0.75)	Medium (0.5)

An Expected core is calculated from the estimated knowledge of the Quiz item difficulty and the Quiz Taker Proficiency
Expected_Score = (P – D + 1)/2

To speed up computation, floating point arithmetic can be avoided
Fuzzy Arithmmetic can operate between Levels (0, 10000)

Residuals Drive the Optimization Process



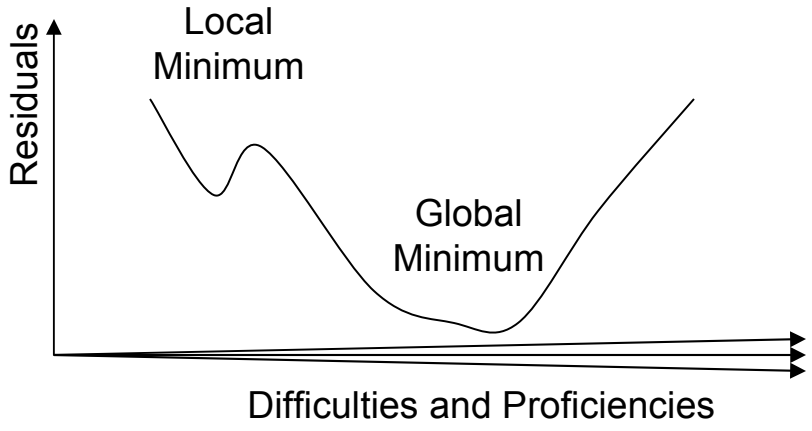
A Residual is defined as the error between
a Measured Scores and an Expected Scores
Residual = Measured_Score - Expected_Score

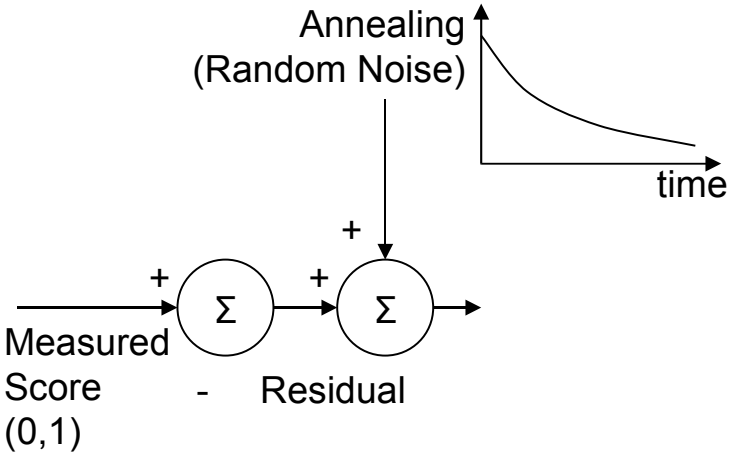
1) Residual = 0 : Expected Score matches Measured Score.
No change required in Difficulty and Proficiency

2) Residual High:
Measured Score is Higher than Expected Score implies that
Estimated Difficulty is too high or
Estimated Proficiency is too low or
both.
Estimated Difficulty must be lowered,
Estimated Proficiency must be raised

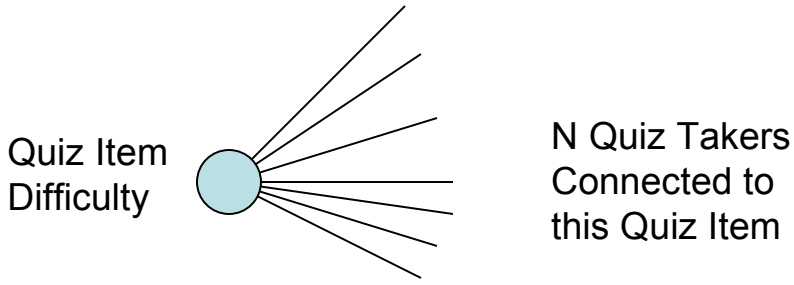
3) Residual Low:
Measured Score is Lower than Expected Score implies that
Estimated Difficulty is too low or
Estimated Proficiency is too high or
both.
Estimated Difficulty must be raised,
Estimated Proficiency must be lowered

Simulated Annealing Overcomes Local Optima



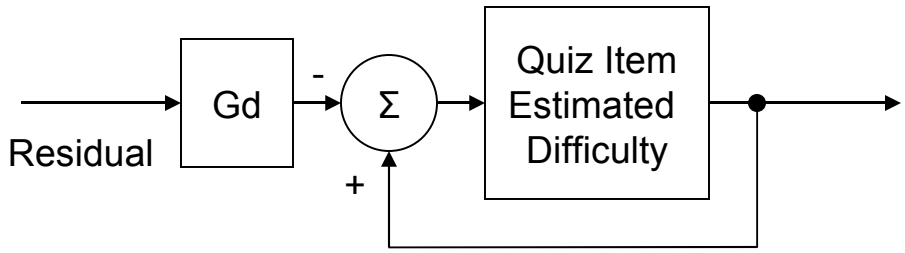


Difficulties and Proficiencies: Essentially Calculated by taking Averages



Batch Method

Quiz_Item_Difficulty = - Sum (Residuals) / N

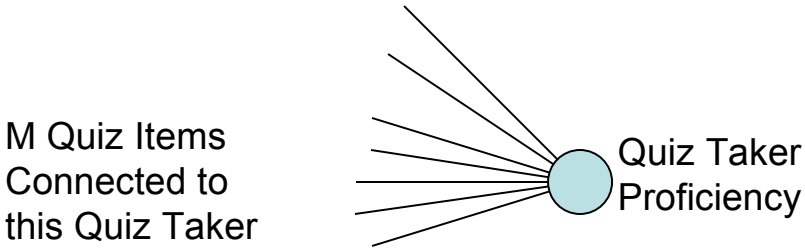


Iterative Method

Quiz_Item_Difficulty = 0
for K = 1 to N:
    Quiz_Item_Difficulty = Quiz_Item_Difficulty - (Residual) / K

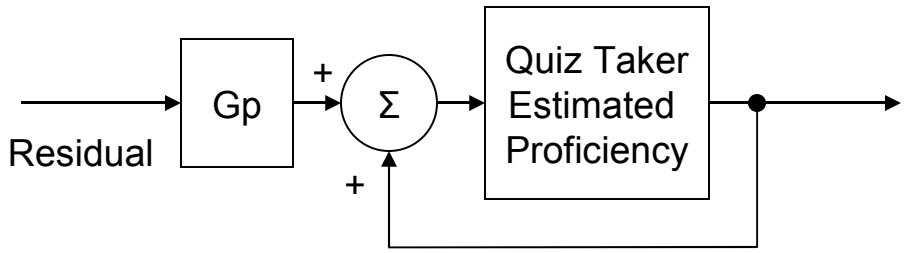
As can be seen the weights are 1/K
(Computation of Kalman Covariance Matrix is simplified)

Difficulty and Proficiency are Calculated by taking Averages



Batch Method

Quiz_Taker_Proficiency = Sum (Residuals) / M

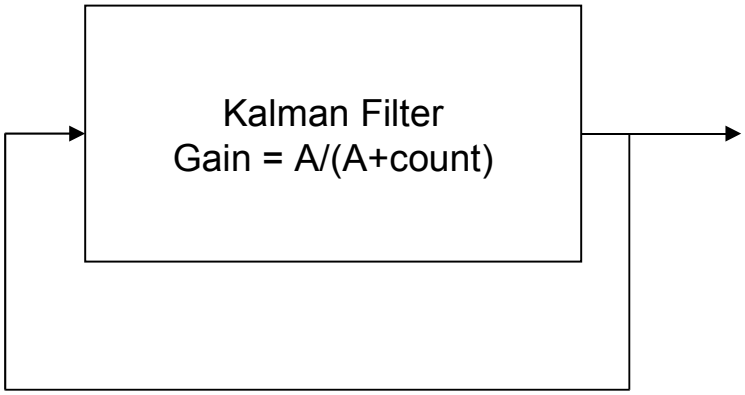


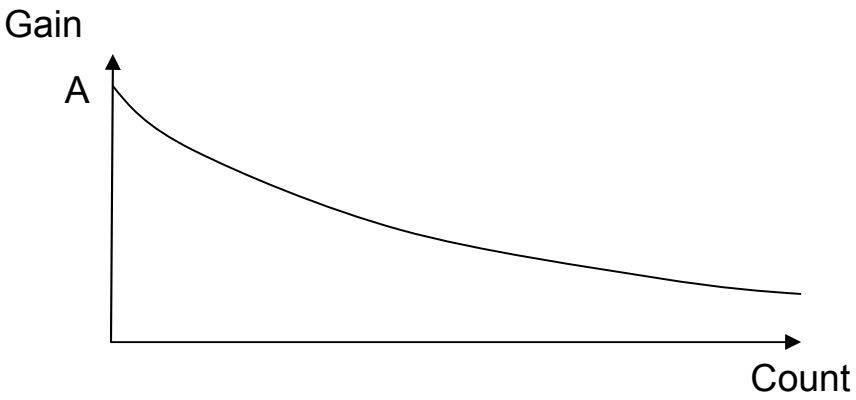
Iterative Method

Quiz_Taker_Proficiency = 0
for L = 1 to M:
Quiz_Taker_Proficiency = Quiz_Taker_Proficiency + (Residual) / L

As can be seen the weights are 1/L
(Computation of Kalman Covariance Matrix is simplified)

Kalman Filter Iteration Convergence





The whole set of score data is repeatedly applied to the Kalman Filter, every time using a smaller gain.