function [best\_feature\_val] = optimizeFeature(feature, LB, UB, step, fin\_model\_low, fin\_model\_med, fin\_model\_high, crit\_model, movie, objParams)

%This function outputs the optimal feature value for a given feature

% The entire movie construction vector (i.e. the "movie" parameter) is

% fixed and only one feature is varied. The value for this feature that

% achieves the optimal objective value is returned (along with the

% objective value function value itself)

%Start by setting the highest value obj value achieved at 0

highest\_value = -Inf;

%Start by setting the value of the feature at its lower bound

best\_feature\_val = LB;

for i=LB:step:UB

%Modify the feature value in the movie vector

movie(1, feature) = {i};

%Run the random forest model for box office

%Need to direct to different models depending on the budget amount:

if movie{1, 'AdjustedBudget'} < 11

box\_office = predict(fin\_model\_low, movie);

elseif movie{1, 'AdjustedBudget'} >= 11 && movie{1, 'AdjustedBudget'} <= 75

box\_office = predict(fin\_model\_med, movie);

else

box\_office = predict(fin\_model\_high, movie);

end

%Run the random forest model for metacritic:

critic = predict(crit\_model, movie);

%Get the objective function value:

obj\_value = getObjective(box\_office, critic, objParams);

obj\_value

%if this objective function value is the best so far, update:

if obj\_value > highest\_value

highest\_value = obj\_value;

best\_feature\_val = i;

end

end

end