Experiment to see how varying the values of attributes can influence the classifier score (a value of 0.5 or more means that the decision tree predictive model expects the student to get a mark of 10 or more).

In [1]:

```
data_file = r'Student_Prepared_FirstTerm.csv'  # Student_Prepared_FirstTerm.csv or Studen
# CHOOSE A PERSONA: "me", "wild-child", "middle-class"
persona = "wild-child"  # "me", "middle-class"
```

Contents [-]

In [2]:

```
%run "Prep Student Classifier.ipynb"
```

In [3]:

```
from ipywidgets import interact, interact_manual
import ipywidgets as widgets
```

In [4]:

```
# function to return a dict of widgets that may be passed to interact() using the **kwargs
def make_widgets(d, cat_cols, init_vals):  # d will be fs_data
    w_dict = dict()
    for c in d.columns:
        if c in cat_cols:
            w_dict[c] = widgets.Dropdown(options=list(d[c].unique()), value=init_vals[c])
        else:
            w_dict[c] = widgets.IntSlider(min=min(d[c]), max=max(d[c]), value=init_vals[c
    return w_dict
```

In [5]:

```
def f(**kwargs):
    print "Classifier score:", predict_1(kwargs)
```

These are for only those features with non-zero importance according to the trained classifier. Not in order (because make_widgets returns a dict).

In [6]:

```
interact(f, **make_widgets(fs_data[undummied_nz_importances.index], cat_cols, init_vals=p
```

schoolsup: no
school: GP
absences: 20
guardian: father
age: 17
studytime: 1
sex: F
Fedu: 1
Medu: 2
failures: 2
Subject: Maths
higher: no

```
Classifier score: 0.0412281104932
```

In [ ]: