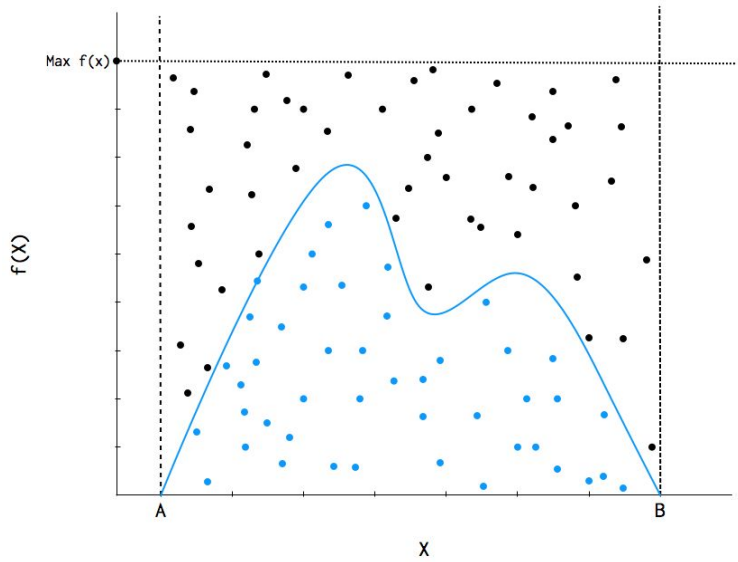
Spark Assignment

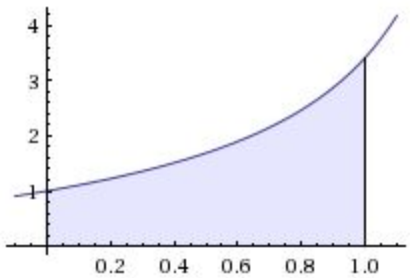
Monte Carlo integration

Large dataset analysis is the main use case of Spark. However, Spark can be used to perform compute intensive tasks as well. Numerical integration is a good example problem, as it falls in this use case class.



The **Monte Carlo integration** method, is a way to get an approximation of the definite integral of a function *f(x)*, over an interval *[A,B]*. Given a value $Max_{f(x)}$, which *f(x)* never exceeds, we first randomly draw *N* uniformly distributed points $(x_1,y_1) \dots (x_N,y_N)$ *s.t.* $x_1 \dots x_N \in [A,B]$, $y_1 \dots y_N \in [0,Max_{f(x)}]$. Then, assuming that *f(x)* is positive over *[A,B]*, the fraction of points that fell under *f(x)* will be roughly equal to the area under the curve, divided by the total area of the rectangle in which we randomly drew points. Hence, the definite integral of *f(x)* over *[A,B]* is roughly equal to: $(B-A)\,Max_{f(x)}\,n_P / tot_P$, where n_p is the number of points that fell under *f(x)*, and tot_p is the total number of randomly drawn points.

Task write a program in spark to approximate the definite integral of *f(x) = (1 + sin(x)) / cos(x)* over [0,1]. Such function is positive over [0,1], and it is lower than *4*.



For the purpose of this assignment drawing 1000 points is good enough.