Mineral exploration as a game of chance

Bell, Peter N

University of Victoria

12 February 2015

Mineral exploration as a game of chance

Peter N. Bell

Department of Economics, University of Victoria

Author note

Contact information: pnbell@uvic.ca, 250 588 6939.

Mailing address: BEC 360, 3800 Finnerty Road (Ring Road), Victoria, BC V8P 5C2 Canada.

Abstract

Exploration is a costly activity that helps a business improve their understanding of a potential mineral deposit. Yet, even with strong exploration results, the business faces uncertainty over the value of the mine. I model this situation as a game of chance. The game starts by giving an agent an asset with random value and ends when the agent chooses to accept the random value or reject it and receive zero instead. The agent can pay to learn more about the asset's value as many times as they like before they end the game, but no amount of exploration will remove all uncertainty. I provide a decision rule for the agent based on an interval estimate for the asset value and analyze performance of the decision rule in a simulation experiment.

Keywords: Mineral exploration, game theory, learning, simulation.

JEL Classification: C02, C44, C63, C70, D83, Q39.

Mineral exploration as a game of chance

1. Introduction

Exploration helps a mining company determine the potential value of a mineral deposit, but a mine may not be as profitable in production as expected from exploration results. Any number of things could be at fault, such as changes in commodity prices or problems with mine operations. However, the fundamental situation is of interest to economists: a company pays for information about the value of a deposit then decides whether to develop the deposit or not. I describe this as a problem of decision making under uncertainty with learning and analyze it as a game of chance.

Although there is extensive research on mineral exploration by economists, the standard approaches do not model the problem as a game of chance (Cairns, 1990). Pindyck (1980) and Arrow and Chang (1982) are amongst the first authors to include uncertainty in models of mineral exploration. They use dynamic programming to solve general equilibrium models, which have appealing theoretical properties but generate some unrealistic results (Cairns, 1990, p.369). Cairns and Quyen (1998) contribute to this literature with a model that allows for learning about deposits based on spatial correlation. Furthermore, Cairns and Quyen discuss their results in terms of a tradeoff between exploitation of existing deposits and exploration for new ones, which is a fundamental concept in mineral economics (Adelman, 1970).

The tradeoff between exploitation and exploration also appears in the multi-arm bandit problem, which is important in computational science and machine learning (Brown & Smith, 2013). In the multi-arm bandit problem, an agent chooses how to play several different slot machines with unknown odds; the agent faces a tradeoff between exploiting machines with good odds and exploring for machines with better odds. This is a classic example of decision making under uncertainty with learning. Researchers in decision theory have used the multi-armed bandit problem to develop industrial strength tools for mineral exploration, such as Bickel and Smith (2006). Bickel and Smith determine the optimal order to drill six different holes depending on the success or failure in each hole. They treat exploration as a sequential process that can exploit spatial correlation of drill holes, which is particularly well suited to exploration in oil and gas deposits.

My approach to the topic is unique in the research literature because I model exploration as a game of chance. The game is simple: nature offers an economic agent a contract with random value, then the agent pays for information about that value until they are ready to accept or reject the contract. My model does not have the standard features of an economic model, such as measures of economic welfare or resource scarcity, nor standard features from decision theory, such as spatial correlation (Bickel and Smith, 2006). Instead, I aim to introduce a toy model that can reveal important aspects of decision making under uncertainty with learning.

Furthermore, my approach is unique because I analyze the toy model using tools from computational economics. In particular, I suppose that the agent acts according to a particular decision rule rather than utility maximization. This approach can be controversial because it is not necessarily optimal or rational, but it is influential in computational economics. Hommes (2013) provides a comprehensive review of the value of this approach in modeling financial markets. Hommes shows that simple decision rules can produce a variety of phenomenon in simulation that match stylized empirical facts, and even match patterns observed in experiments better than standard economic models. My paper aims to promote the use of computational economics for modeling business decision making in mineral exploration.

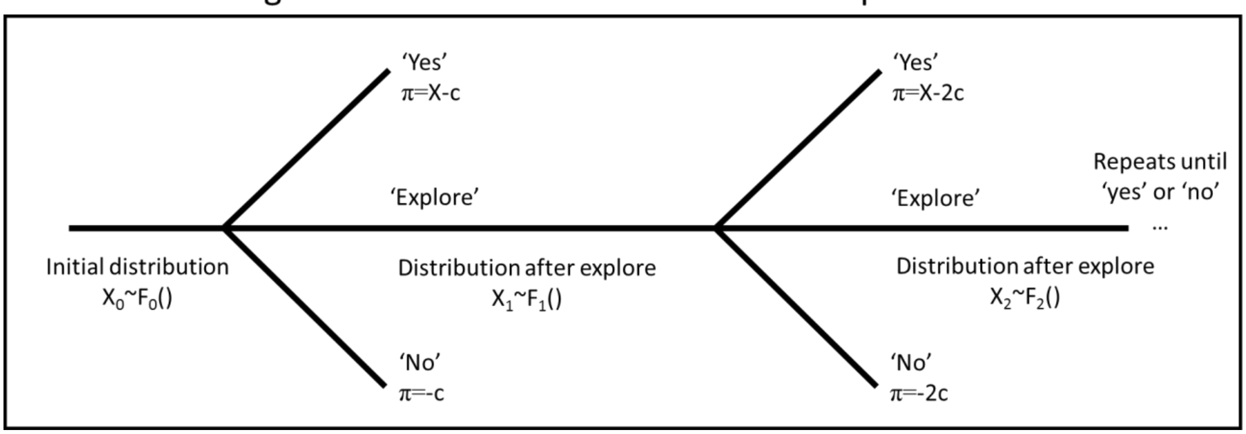
2. Model

2.1 General Model

I introduce the mineral exploration game in Figure 1. Nature starts the game by drawing a random value X from the initial distribution $F_0()$. The agent knows the initial distribution, but not the particular value of X. At each round of the game, the agent can choose yes, no, or explore. The game ends when the agent picks yes or no. If they pick yes, then they get X. If they pick no, then they get zero with certainty. Since the value X can be greater or less than zero, there is potential for risk or reward in picking yes versus no.

I denote the total number of times that the agent explores before they decide as n. Each round of exploration costs the agent c, but allows them to update their assessment of the probability distribution for the value to the posterior distribution $X_i \sim F_i()$. I denote the net profit that the agent earns at the end of the game as π. If the agent picks yes, then π=X-nc. If no, then π=-nc. I do not include a time value of money.

Figure 1 – Decision tree for model of exploration



I analyze the model by assuming the agent uses a decision rule at each stage of the game. The decision rule is based on a vector of statistics Θ_i calculated from the distribution $F_i()$. I define a set of values for the statistics that causes the agent to choose yes Θ_Y, no Θ_N, or explore Θ_E. The decision rule is simply: if Θ_i in Θ_Y then pick yes; if Θ_i in Θ_N then pick no; else explore again.

In this paper I use a decision rule based on an interval estimate of the value X. Let $\Theta_i=\{Q_L,Q_U\}$ where Q_L is the lower quartile, $F_i(X\leq Q_L)=0.25$, and Q_U is the upper quartile, $F_i(X\leq Q_U)=0.75$. I define the set $\Theta_Y=\{Q_L,Q_U:0<Q_L<Q_U\}$, $\Theta_N=\{Q_L,Q_U:Q_L<Q_U<0\}$, and $\Theta_E=\{Q_L,Q_U:Q_L<0<Q_U\}$. According to this decision rule, the agent picks yes when the lower quartiles is positive or, in other words, when X is positive at 75% confidence level. The agent picks no when X is negative with 75% confidence level. The agent chooses to explore again when neither condition is satisfied or, in other words, when the 50% confidence interval for X contains zero.

2.2 Probability model

In this section I describe how to generate the posterior distribution of value X after one round of exploration. Suppose the initial probability model is $X_0\sim U(a_0,b_0)$. After one round of exploration, the distribution becomes $X_1\sim U(a_1,b_1)$. The new values (a_1,b_1) must satisfy several conditions. First, the new values cannot exclude the true value, $a_1<X<b_1$. Second, the distance between the new values must be 10% smaller than the initial model, as in Equation (1).

(1) $(b_1-a_1)=0.9(b_0-a_0)$

In order for the new values (a_1,b_1) to tighten the distribution around the true value, they must increase the lower bound $a_0<a_1$ and decrease the upper bound $b_1<b_0$. I ensure this by constructing the new values as in Equation (2), which include two new variables $e_a,e_b>0$.

(2) $a_1=a_0+e_a$; $b_1=b_0-e_b$

I combine Equations (1) and (2) to get a constraint on the values of e_a and e_b in Equation (3).

(3) $e_a+e_b=0.1(b_0-a_0)$

I draw a random value for e_a from uniform distribution, $e_a\sim U(0,0.1(b_0-a_0))$, then calculate e_b from Equation (3). Based on e_a and e_b, I calculate (a_1,b_1) from Equation (2) and check the condition $a_1<X< b_1$. This procedure ensures that the new distribution $U(a_1,b_1)$ will be more tightly concentrated around true value X. I assume the agent uses the new distribution $X_1\sim U(a_1,b_1)$ to make their next decision. The agent calculates the statistics Θ_1 from the new distribution and determines if the agent is ready to make a decision yet. If Θ_1 in $\Theta_{Y,}$ then they pick yes. If Θ_1 in Θ_N, then they pick no. Else, they do another round of exploration and update the distribution again based on the procedure described this section.

2.3 Results

I specify the parameters in the model to ensure that the cost of exploration is low relative to the potential value of the mine, cost c=0.01 and initial bounds a_0=-1, b_0=1. This means the true value has a uniform distribution, $X\sim U(-1,1)$. I allow the agent to play the game 100,000 times and calculate the distribution for key variables associated with the decision rule.

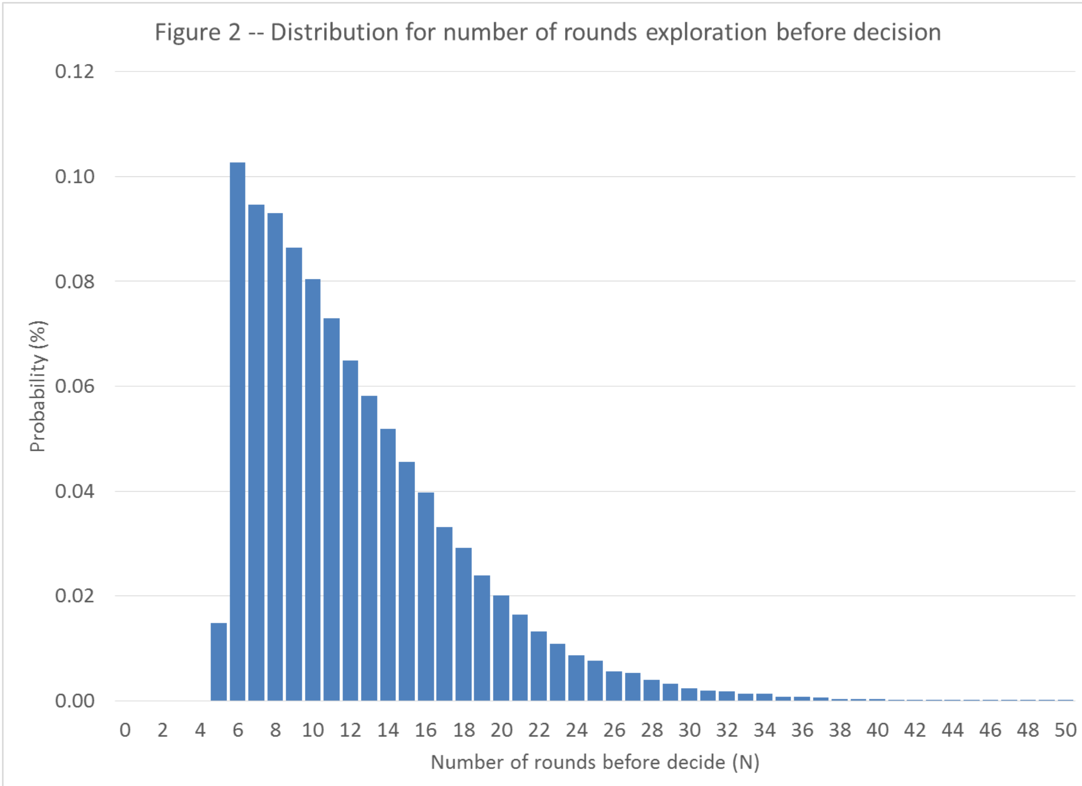


Figure 2 show the distribution for the number of times that the agent chooses exploration before they pick yes or no. It shows that the agent needs at least five rounds of exploration before they make a decision, which is expected based on the construction of the model and in line with industry standards for mineral exploration. The figure also shows that the agent sometimes need as many as 50 rounds to decide! The long tail for this distribution is due to the fact that the quantile-based decision rule has a hard time distinguishing the sign of small values: if X is close to zero, then the confidence interval must be very small before it will be entirely above or below zero. Thus, it may be optimal for the agent to give up on projects after many rounds of exploration because the value X is likely to be small.

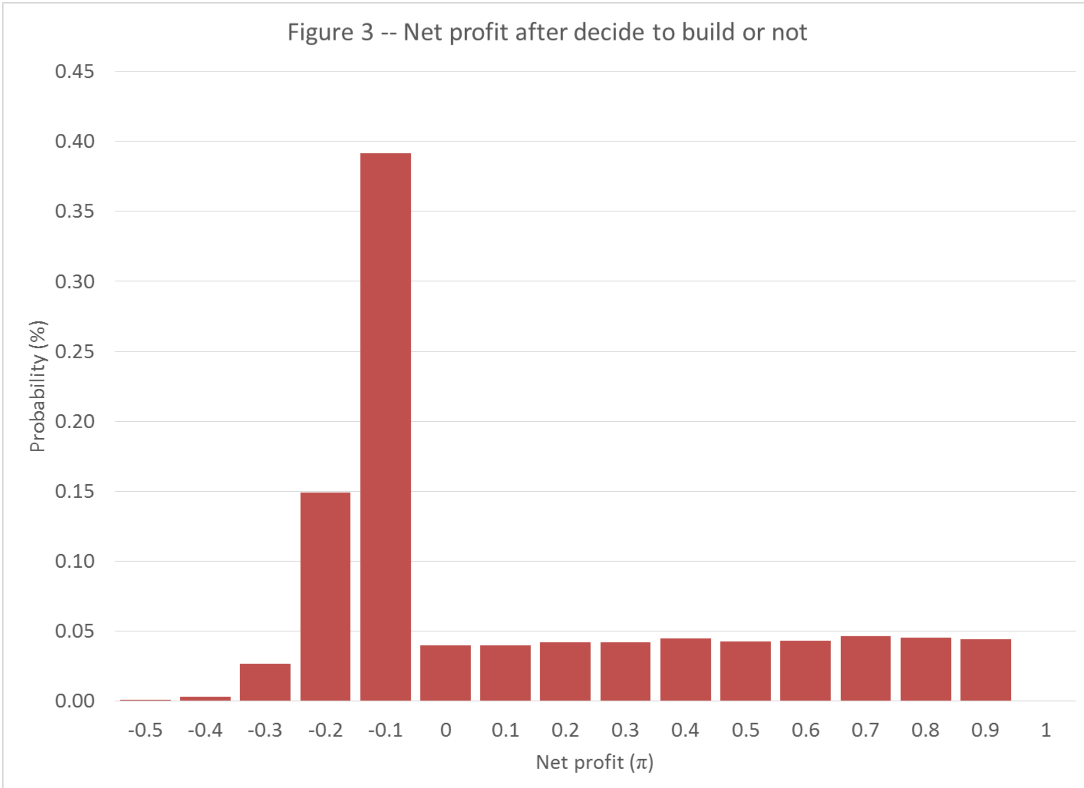


Figure 3 shows the distribution for net profit π. The figure shows that profits have an asymmetric distribution with positive skew. The large probability of large positive profits and the small probability of large negative profits is a desirable feature. However, the large probability of small negative profits shows that there is some risk associated with the exploration game.

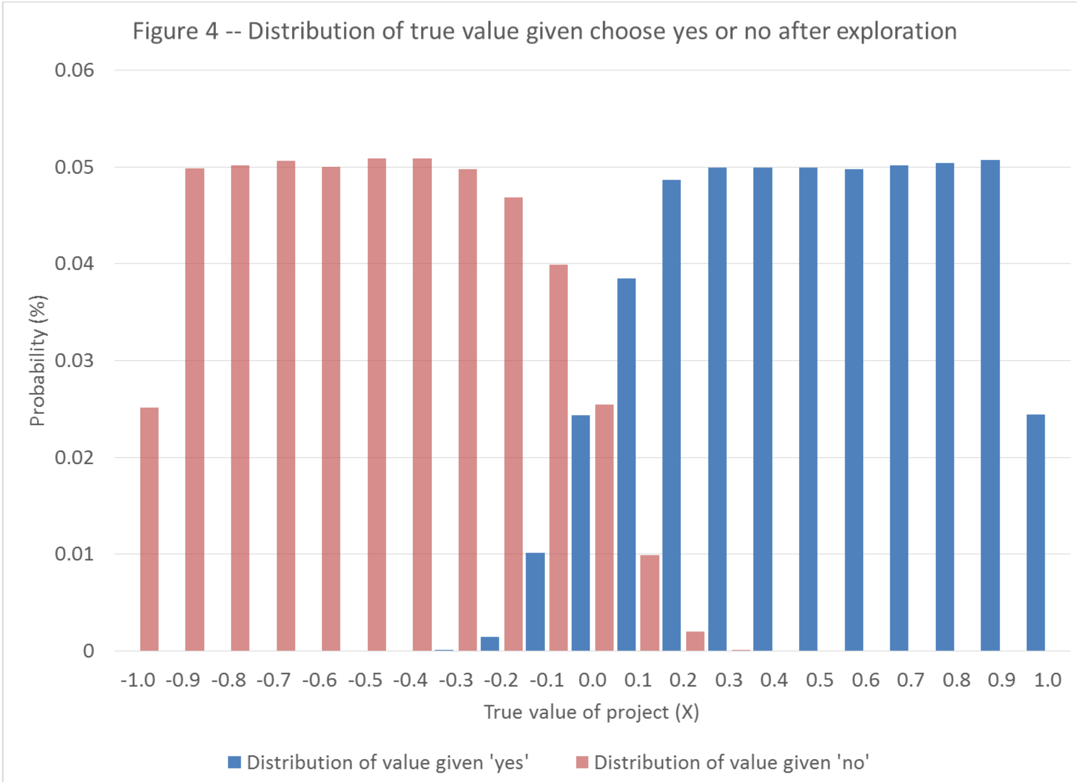


Figure 4 shows the conditional distribution of the true value of X given that the agent picks yes or no at the end of the game. The symmetry in the two distributions is due to the symmetry in the agent's decision rule. The two distributions overlap around zero because it is difficult for the quantile-based decision rule to distinguish between small positive or negative values. The results also show that the decision rule is able to effectively distinguish between large positive or negative values for X.

3. Discussion

In this paper I describe mineral exploration as a game of chance, where an agent can pay for information to improve their knowledge of the value of a mineral deposit. I introduce a decision rule to determine when the agent stops exploring and decides to build the mine or not. I explore the behavior of the decision rule in simulation and find that the decision rule has desirable features, such as positive skew in the distribution of profits.

The model can be changed in several ways. For one, it is possible to change the probability model for the true value to allow for correlation across different rounds of the game. For another, it is possible to compare different versions of the decision rule; if the decisions were based on the 25% and 95% quantiles, then the agent would require higher confidence to pick yes and have higher accuracy when the true value is positive. An interested reader could even compare different decision rules based on some measure over the distribution of profit, such as a utility function. Finally, it is possible to introduce different parties, such as management and financier. If these two parties update their distributions about the value in different ways after each round of exploration, then the model could provide a way to explore principal-agent problems in context of decision making under uncertainty with learning.

Acknowledgements:

This research was supported by the Joseph-Armand Bombardier Canada Graduate Scholarship – Doctoral from the Social Sciences and Humanities Research Council of Canada.

References

Adelman, M.A. (1970). Economics of Exploration for Petroleum and Other Minerals. *Geoexploration, 8*, 131—150.

Arrow, K.J., & Chang, S. (1982). Optimal Pricing, Use and Exploration of Uncertain Natural Resource Stocks. *Journal of Environmental Economics and Management, 9*, 1—10.

Bickel, J.E., & Smith, J.E. (2006). Optimal sequential exploration: A binary learning model. *Decision Analysis, 3*(1), 16–32

Brown, D.B. & Smith, J.E. (2013). Optimal Sequential Exploration: Bandits, Clairvoyants, and Wildcats. *Operations Research, 61*(3), 644—665.

Cairns, R.D. (1990). The economics of exploration for non-renewable resources. *Journal of Economic Surveys, 4*(4), 361—395.

Cairns, R.D., & Quyen, N.V. (1998). Optimal Exploration for and Exploitation of Heterogeneous Mineral Deposits. *Journal of Environmental Economics and Management, 35*, 164–189.

Hendricks, K. & Kovenock, D. (1989). Asymmetric Information, Informational Externalities and Efficiency: The Case of Oil Exploration. *Rand Journal of Economics 20*(2), 164—182.

Hommes, C. (2013). *Behavioral Rationality and Heterogeneous Expectations in Complex Economic Systems*. Cambridge, UK: Cambridge University Press.

Lin, C.C. (2009). Estimating strategic interactions in petroleum exploration. *Energy Economics, 31*, 586—594.

Pindyck, R.S. (1980). Uncertainty and Exhaustible Resource Markets. *Journal of Political Economy, 88*(6), 1203—1225.

Appendix

```
%%  Code Appendix - Mineral Exploration game
%   Written by Peter Bell, February 5 2014
%
%
%%  Section 1:  Specify global parameters
%
clear all;
numLoop=0;  cost=1/100;  aTrue=-1;  bTrue=1;    decisionVar=0;
xTrue=aTrue+(bTrue-aTrue)*rand();

%%  Section 2:  Loop over many rounds of exploration
numGames=100000;
for iGame=1:numGames
    numLoop=0;  decisionVar=0;
    xTrue=aTrue+(bTrue-aTrue)*rand();
    aLoop=aTrue;    bLoop=bTrue;
    while decisionVar==0
        numLoop=numLoop+1;
        quantLow=aLoop+0.25*(bLoop-aLoop);
        quantUp=aLoop+0.75*(bLoop-aLoop);

        if 0<quantLow & 0<quantUp
            decisionVar=1;
            netValue=xTrue-numLoop*cost;
        elseif quantLow<0 & quantUp<0
            decisionVar=-1;
            netValue=0-numLoop*cost;
        end

        containTrue=0;
        while containTrue==0
            if (xTrue-aLoop)<0.00001
                containTrue=1;
                aLoop=aLoop;
                bLoop=bLoop-0.1;
            elseif (bLoop-xTrue)<0.0001
                containTrue=1;
                aLoop=aLoop+0.1;
                bLoop=bLoop;
            else
                changeA=0.1*(bLoop-aLoop)*rand();
                changeB=0.1*(bLoop-aLoop)-changeA;
                aLoopTemp=aLoop+changeA;
                bLoopTemp=bLoop-changeB;
                if aLoopTemp<xTrue & xTrue<bLoopTemp
                    containTrue=1;
                    aLoop=aLoopTemp;
                    bLoop=bLoopTemp;
                end
            end
        end
    end
    results(iGame,:)=[numLoop decisionVar netValue xTrue];
    iGame
```

```
end

%%  Section 3:  Save results for figures
[numLoopHist loopTicks]=hist(results(:,1),0:1:50);
saveOne=[numLoopHist' loopTicks'];
save('tables1.txt','saveOne','-ascii')

[decisionHist decTicks]=hist(results(:,2),-1:1:1);
saveTwo=[decisionHist' decTicks'];
save('tables2.txt','saveTwo','-ascii')

[netHist netTicks]=hist(results(:,3),-0.5:0.1:1);
saveThree=[netHist' netTicks'];
save('tables3.txt','saveThree','-ascii')

conditYesLogic=results(:,2)==1;
conditYes=results(conditYesLogic,:);
[conditYesHist yesTicks]=hist(conditYes(:,4),-1:0.1:1);
saveFour=[conditYesHist' yesTicks'];
save('tables4.txt','saveFour','-ascii')


conditNoLogic=results(:,2)==-1;
conditNo=results(conditNoLogic,:);
[conditNoHist noTicks]=hist(conditNo(:,4),-1:0.1:1);
saveFive=[conditNoHist' noTicks'];
save('tables5.txt','saveFive','-ascii')
```