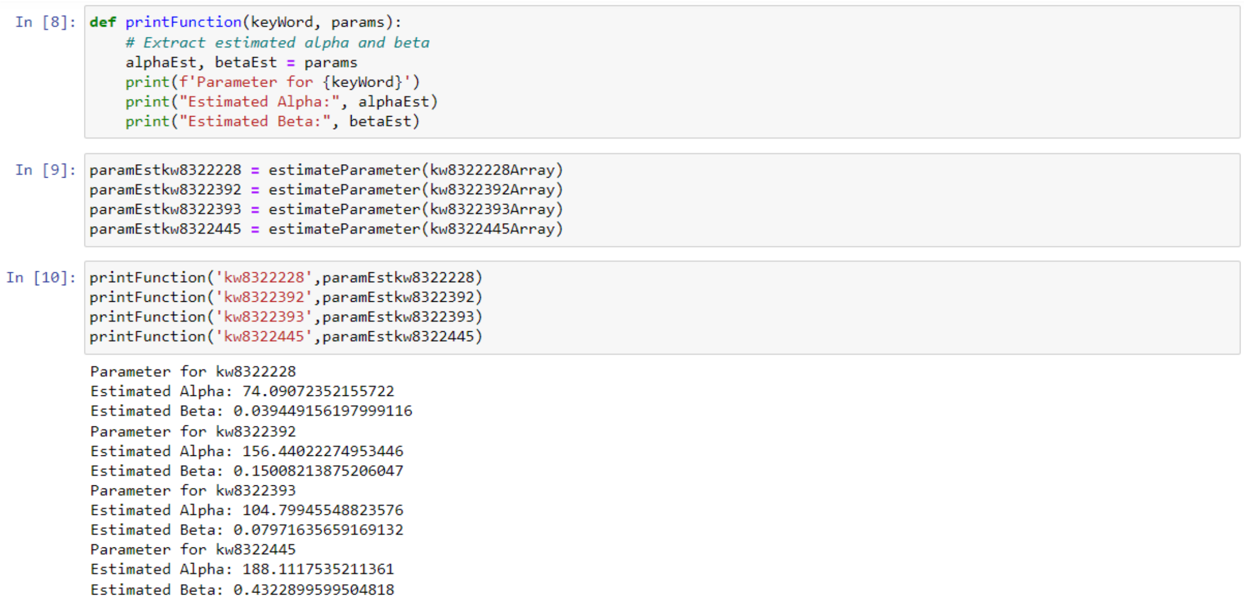
Paid Search Bid Optimization and Display Advertising

Contents

Question 1 Paid Search Bid Optimization

Part A: Estimate the alpha and beta parameters for each of these four keywords Solution



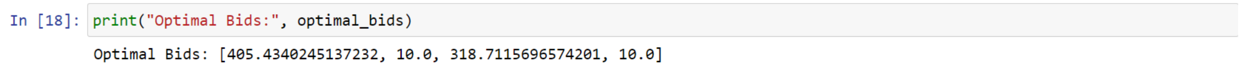
```
In [8]:  def printFunction(keyWord, params):
             # Extract estimated alpha and beta
             alphaEst, betaEst = params
             print(f'Parameter for {keyWord}')
             print("Estimated Alpha:", alphaEst)
             print("Estimated Beta:", betaEst)

In [9]:  paramEstkw8322228 = estimateParameter(kw8322228Array)
         paramEstkw8322392 = estimateParameter(kw8322392Array)
         paramEstkw8322393 = estimateParameter(kw8322393Array)
         paramEstkw8322445 = estimateParameter(kw8322445Array)

In [10]: printFunction('kw8322228',paramEstkw8322228)
         printFunction('kw8322392',paramEstkw8322392)
         printFunction('kw8322393',paramEstkw8322393)
         printFunction('kw8322445',paramEstkw8322445)

         Parameter for kw8322228
         Estimated Alpha: 74.09072352155722
         Estimated Beta: 0.039449156197999116
         Parameter for kw8322392
         Estimated Alpha: 156.44022274953446
         Estimated Beta: 0.15008213875206047
         Parameter for kw8322393
         Estimated Alpha: 104.79945548823576
         Estimated Beta: 0.07971635659169132
         Parameter for kw8322445
         Estimated Alpha: 188.1117535211361
         Estimated Beta: 0.4322899599504818
```

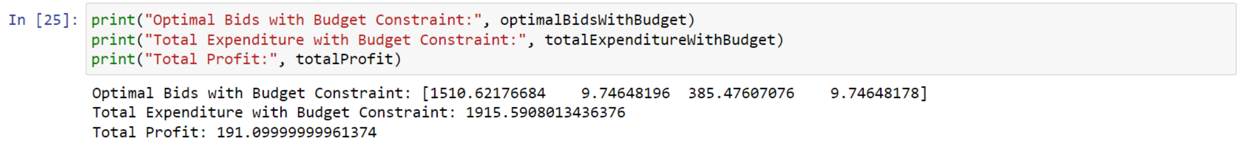
Part B: Optimal bids for each of the four keywords.



```
In [18]: print("Optimal Bids:", optimal_bids)

         Optimal Bids: [405.4340245137232, 10.0, 318.7115696574201, 10.0]
```

Part C: With budget constraint, USD 3000 across 4 key words, optimal bid value, corresponding profit, corresponding total expenditure

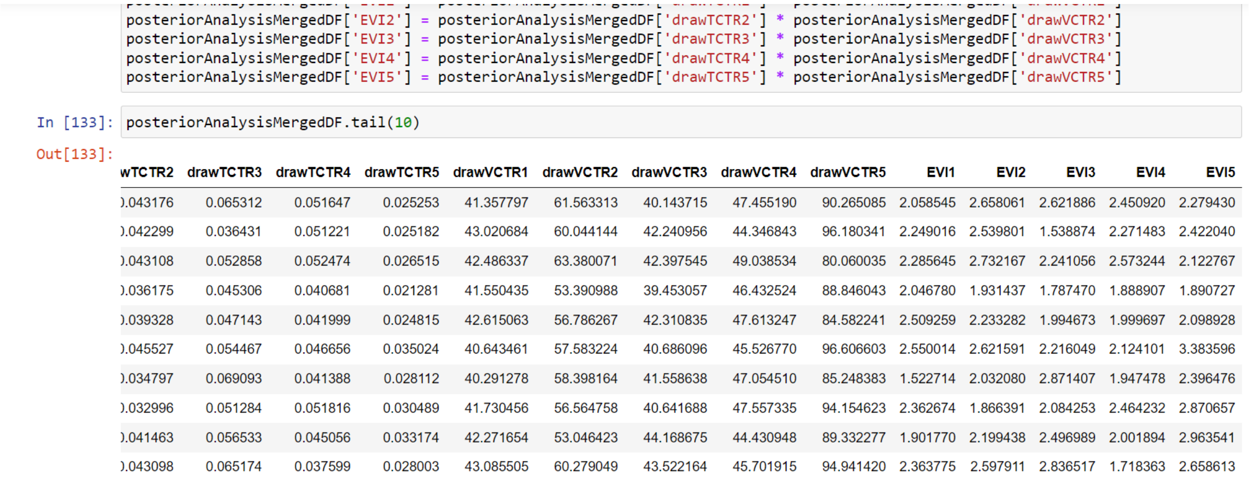


```
In [25]: print("Optimal Bids with Budget Constraint:", optimalBidsWithBudget)
         print("Total Expenditure with Budget Constraint:", totalExpenditureWithBudget)
         print("Total Profit:", totalProfit)

         Optimal Bids with Budget Constraint: [1510.62176684    9.74648196  385.47607076    9.74648178]
         Total Expenditure with Budget Constraint: 1915.5908013436376
         Total Profit: 191.09999999961374
```

Question 2 Display Advertising Assessment

Question: Assess for each campaign, also of its true expected volume per exposure or impression (abbreviated as "EVI")

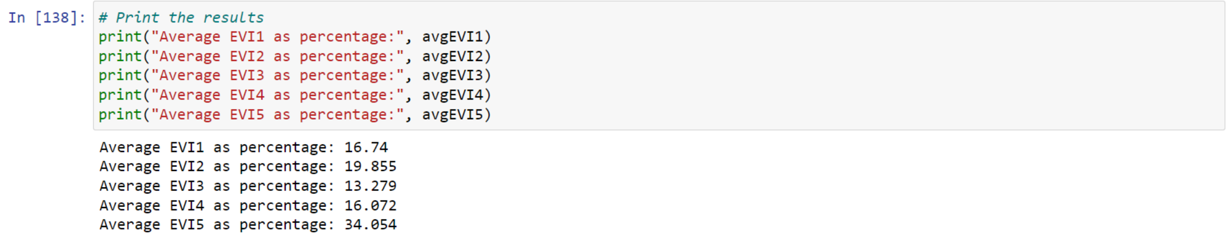


```
posteriorAnalysisMergedDF['EVI2'] = posteriorAnalysisMergedDF['drawTCTR2'] * posteriorAnalysisMergedDF['drawVCTR2']
posteriorAnalysisMergedDF['EVI3'] = posteriorAnalysisMergedDF['drawTCTR3'] * posteriorAnalysisMergedDF['drawVCTR3']
posteriorAnalysisMergedDF['EVI4'] = posteriorAnalysisMergedDF['drawTCTR4'] * posteriorAnalysisMergedDF['drawVCTR4']
posteriorAnalysisMergedDF['EVI5'] = posteriorAnalysisMergedDF['drawTCTR5'] * posteriorAnalysisMergedDF['drawVCTR5']
```

```
In [133]: posteriorAnalysisMergedDF.tail(10)
```

Out[133]:

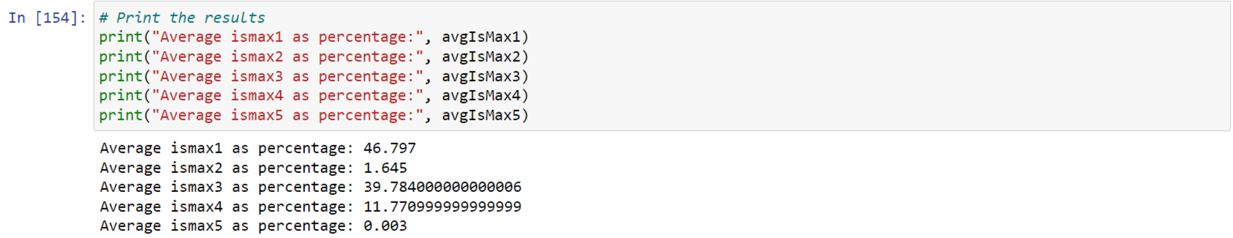
wTCTR2	drawTCTR3	drawTCTR4	drawTCTR5	drawVCTR1	drawVCTR2	drawVCTR3	drawVCTR4	drawVCTR5	EVI1	EVI2	EVI3	EVI4	EVI5
.043176	0.065312	0.051647	0.025253	41.357797	61.563313	40.143715	47.455190	90.265085	2.058545	2.658061	2.621886	2.450920	2.279430
.042299	0.036431	0.051221	0.025182	43.020684	60.044144	42.240956	44.346843	96.180341	2.249016	2.539801	1.538874	2.271483	2.422040
.043108	0.052858	0.052474	0.026515	42.486337	63.380071	42.397545	49.038534	80.060035	2.285645	2.732167	2.241056	2.573244	2.122767
.036175	0.045306	0.040681	0.021281	41.550435	53.390988	39.453057	46.432524	88.846043	2.046780	1.931437	1.787470	1.888907	1.890727
.039328	0.047143	0.041999	0.024815	42.615063	56.786267	42.310835	47.613247	84.582241	2.509259	2.233282	1.994673	1.999697	2.098928
.045527	0.054467	0.046656	0.035024	40.643461	57.583224	40.686096	45.526770	96.606603	2.550014	2.621591	2.216049	2.124101	3.383596
.034797	0.069093	0.041388	0.028112	40.291278	58.398164	41.558638	47.054510	85.248383	1.522714	2.032080	2.871407	1.947478	2.396476
.032996	0.051284	0.051816	0.030489	41.730456	56.564758	40.641688	47.557335	94.154623	2.362674	1.866391	2.084253	2.464232	2.870657
.041463	0.056533	0.045056	0.033174	42.271654	53.046423	44.168675	44.430948	89.332277	1.901770	2.199438	2.496989	2.001894	2.963541
.043098	0.065174	0.037599	0.028003	43.085505	60.279049	43.522164	45.701915	94.941420	2.363775	2.597911	2.836517	1.718363	2.658613



```
In [138]: # Print the results
          print("Average EVI1 as percentage:", avgEVI1)
          print("Average EVI2 as percentage:", avgEVI2)
          print("Average EVI3 as percentage:", avgEVI3)
          print("Average EVI4 as percentage:", avgEVI4)
          print("Average EVI5 as percentage:", avgEVI5)

          Average EVI1 as percentage: 16.74
          Average EVI2 as percentage: 19.855
          Average EVI3 as percentage: 13.279
          Average EVI4 as percentage: 16.072
          Average EVI5 as percentage: 34.054
```

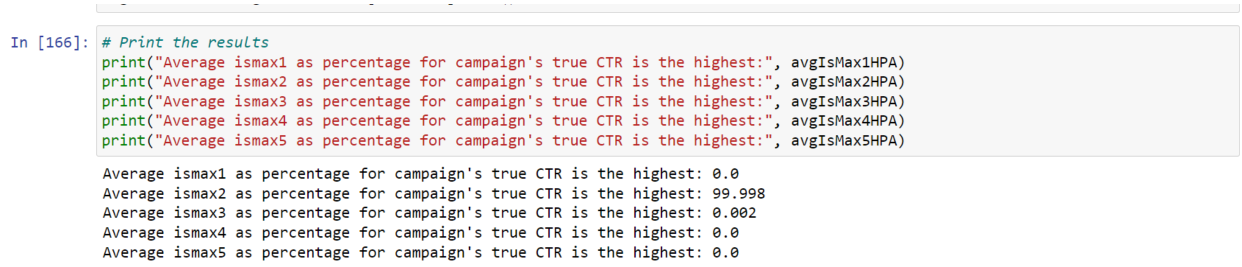
2.1 Compute the Bayesian posterior probability that the campaign's true click-through-rate is the highest across all campaigns.



```
In [154]: # Print the results
          print("Average ismax1 as percentage:", avgIsMax1)
          print("Average ismax2 as percentage:", avgIsMax2)
          print("Average ismax3 as percentage:", avgIsMax3)
          print("Average ismax4 as percentage:", avgIsMax4)
          print("Average ismax5 as percentage:", avgIsMax5)

          Average ismax1 as percentage: 46.797
          Average ismax2 as percentage: 1.645
          Average ismax3 as percentage: 39.784000000000006
          Average ismax4 as percentage: 11.770999999999999
          Average ismax5 as percentage: 0.003
```

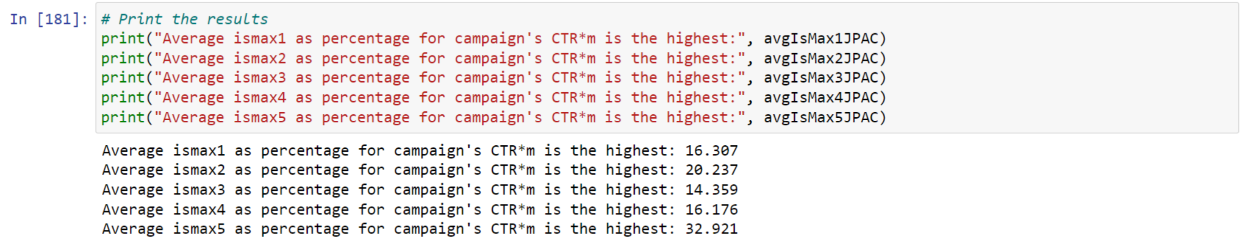
2.2 Compute the Bayesian posterior probability that the campaign's true average post-click volume per click is the highest across all campaigns.



```
In [166]: # Print the results
          print("Average ismax1 as percentage for campaign's true CTR is the highest:", avgIsMax1HPA)
          print("Average ismax2 as percentage for campaign's true CTR is the highest:", avgIsMax2HPA)
          print("Average ismax3 as percentage for campaign's true CTR is the highest:", avgIsMax3HPA)
          print("Average ismax4 as percentage for campaign's true CTR is the highest:", avgIsMax4HPA)
          print("Average ismax5 as percentage for campaign's true CTR is the highest:", avgIsMax5HPA)

          Average ismax1 as percentage for campaign's true CTR is the highest: 0.0
          Average ismax2 as percentage for campaign's true CTR is the highest: 99.998
          Average ismax3 as percentage for campaign's true CTR is the highest: 0.002
          Average ismax4 as percentage for campaign's true CTR is the highest: 0.0
          Average ismax5 as percentage for campaign's true CTR is the highest: 0.0
```

2.3 Compute the Bayesian posterior probability that the campaign's true expected volume per exposure (impression) is the highest.



```
In [181]: # Print the results
          print("Average ismax1 as percentage for campaign's CTR*m is the highest:", avgIsMax1JPAC)
          print("Average ismax2 as percentage for campaign's CTR*m is the highest:", avgIsMax2JPAC)
          print("Average ismax3 as percentage for campaign's CTR*m is the highest:", avgIsMax3JPAC)
          print("Average ismax4 as percentage for campaign's CTR*m is the highest:", avgIsMax4JPAC)
          print("Average ismax5 as percentage for campaign's CTR*m is the highest:", avgIsMax5JPAC)

          Average ismax1 as percentage for campaign's CTR*m is the highest: 16.307
          Average ismax2 as percentage for campaign's CTR*m is the highest: 20.237
          Average ismax3 as percentage for campaign's CTR*m is the highest: 14.359
          Average ismax4 as percentage for campaign's CTR*m is the highest: 16.176
          Average ismax5 as percentage for campaign's CTR*m is the highest: 32.921
```

Appendix

Display Advertising Assessment	
Task 1: Bayesian posterior probability that a certain campaign's true CTR is the highest	DASS1posteriorAnalysis100000.csv
Task 2: Bayesian posterior probability that a certain campaign's average profit volume is the highest	DASS1posteriorAnalysishighCTR100000.csv
Task 3: Bayesian posterior probability that a certain campaign's CTR*m is the highest	DASS1posteriorAnalysisjointCTR100000.csv