A Dynamic Programming approach to False Discovery Rate control in Bayesian Multiple Hypothesis testing

Soma S. Dhavala*, Bani K. Mallick

Dept. of Statistics,
Texas A& M University
College Station, TX, 77840
{soma, bmallick}@stat.tamu.edu

Abstract

We propose an algorithm for controlling different error measures in a Bayesian multiple testing under generic loss functions, including the widely used uniform loss function. We do not make any specific assumptions about the underlying probability model but require that indicator variables for the individual hypothesis are available, as a component of the inference. Given this information, we recast the multiple hypothesis testing problem as a combinatorial optimization and in particular, the 0-1 knapsack problem which can be solved efficiently using a variety of algorithms, both approximate and exact in nature. This connection opens-up many opportunities for leveraging vast literature available on the variants of the knapsack problems. Under this framework, we can maximize a function of the expected number of true positives, while keeping the expected number of false positives within a user-specified capacity bound, such as the False Discovery Rate. Further, when a dynamic programming implementation is chosen, the algorithm produces uniformly optimal sequential decisions as a function of capacity. This feature is particularly useful to stop the algorithm as soon as the desired capacity is met without needing to look-ahead or repeat the same procedure for every capacity all over again.

Keywords: Dynamic programming, Bayesian Multiple hypothesis testing, False discovery rate, 0-1 Knapsack problem, Multiple testing

1. Introduction

Recent advances microarrays and other genomics technologies have spurred the interest in multiple testing procedures. Due to the nature of the problem, one has to perform potentially thousands of test simultaneously with relatively small sample sizes, often referred to as the *large p, small n* problem. It is imperative that, under these settings, having control on the number of incorrect decisions being made is of paramount importance. Naturally, many methods have been proposed to control different errors measures such as the Family-Wise Error Rate (FWER), False Discovery Rate (FDR), positive FDR (pFDR) among others. In large-scale multiple hypothesis testing problems, the FWER is severely conservative. Benjamani and Hotcheberg, in their seminal paper (Benginemini and Hotchberg, 1995), introduced FDR, defined as the expected proportion of false positives among the rejected hypothesis, which offered a practical alternative to controlling decision errors that is less conservative than FWER. Following this work, many attempts have been made to improve/extend/generalize FDR under different conditions; see for example, Genovese and Wasserman (2002), Storey (2003), Sun and Cai (2007), Tang and Zhang (2007), Efron (2008b), Ferkingstad et al. (2008), Sarkar et al (2008), Roquain and van de Wiel (2009), Chen et al. (2009). An an optimal decision process (S_{ODP}) was proposed in Storey (2007), which maximizes expected true positives counts while minimizing expected false positives counts, emphasizing the need to consider compound decision processes as opposed considering individual tests in isolation to others. Decision theoretic approach were also used in Mueller et al. (2004), Mueller et al. (2007), Scott and Berger (2003), and Peña et al (2010) to offer different insights into controlling FDR.

Guindani et al. (2009) showed that their Bayesian version of the ODP (B_{ODP}) approximates S_{ODP} closely, under a non-parametric set-up. Based on this observation, they determine the optimal decision rules by using the procedures developed for S_{ODP} in Storey et al. (2007). They also suggest several extensions to the models by considering different loss functions, motivated from biological studies. However, while the probability model is set-up in the Bayesian framework, inference is carried in the frequentest framework. In particular, they use cluster configuration induced by the Dirichlet Process as a part of the Markov Chain Monte-Carlo (MCMC), and plug-in pooled maximum likelihood estimates in S_{ODP}. It is unclear what their motivation is to follow this circuitous route, but we

*Soma S. Dhavala is a Ph.D student and Bani K. Mallick is Professor in the Dept. of Statistics at Texas A & M University. Their research was supported by National Science Foundation grant DMS 0914951 and by Award Number KUS-CI-016-04, made by King Abdullah University of Science and Technology (KAUST).

suspect that, unavailability of efficient solutions to determine optimal decision rules could be a reason. This is one of the motivations for us to suggest algorithms to find optimal decision rules in a fully Bayesian way without having to bootstrap or perform grid-based searches as described in Mueller et al. (2004). Secondly, as demonstrated in Mueller et al. (2007), loss functions provide flexibility in controlling FDR, with the possibility of improving FDR in specific cases, similar in spirit to test-specific power-functions being used in the Neyman-Pearson schema of Peña et al (2010) and Foster and Stine (2008). However, we believe that lack of efficient algorithms for completing the inference in Bayesian multiple hypothesis testing is a hurdle to be overcome, which we attempt in this paper.

The rest of the paper is organized as follows: In Section 2, we lay down the inferential goals for multiple hypothesis testing from a Bayesian standpoint, utilizing decision theoretic framework. In the next section, we discuss the 0-1 knapsack problem, one of the well-studied combinatorial optimization problems. We recast the multuple hypothesis testing problem as an the 0-1 knapsack problem by making suitable modifications in Section 4. We provide simulation examples along with a discussion in the next section. We conclude with a discussion of the proposed algorithms in Section 6. Pseduocode for the FDR control is provided in the Appendix.

2. Bayesian Multiple hypothesis testing

Let γ_i be the indicator variable associated with the i-th gene, with $\gamma_i = 0$ when the null hypothesis is true (for example, a gene is not differentially expressed). We use the term gene purely for historical reasons. The set-up is equally applicable to other scenarios such as testing edges in graphical models. We do not make any specific assumptions about the underlying probablity model. For example, the indicator variable used could be specified in the product form as in Sharpf et al. (2009) or as a familiar two-component mixture model under parametric setting as in Gottardo et al. (2006), or under semiparametric settings as in Kim et al. (2009). We only require some mild requirements described in Sec. 3 of Mueller et al. (2007) to carry out inferences from the posterior distribution. Our goal is to test the P hypothesis of the following form:

$$H_{0i} : \gamma_i = 0 \text{ Vs } H_{1i} : \gamma_i = 1 \ \forall i = 1, 2, \ldots, P$$

Let $d_i = 1$ ($d_i = 0$) be decition to reject (fail to reject) the null. Then the outcomes from the above hypothesis test can be summarized in the following table:

	Null true	Alternate true	Total
Accept null	U	T	M-R
Reject null	V	S	R = $\sum d_i$
Total	$M - m_1$	$m_1 = \sum \gamma_i$	M

Clearly, we need to control the number of incorrect decisions. Some commonly used measures are:

FWER: E($V > 1$)

FDR: E($V/R|R > 1$)

pFDR: E(V/R)E($R > 1$)

FNR: E($T/M - R$)

among others and the overall goal is to simultaneously minimize a function of V and T, and maximize a function of U and S. A reasonable way to trade-off these conflicting goals is to, maximize V(or T) while keeping U(or S) within manageable limits specified in terms of the error measures defined above. Of course, we do not know them in reality, so we work with their expected values instead. In the above error measures, we penalized incorrect decisions and rewarded correct decisions in every test the same way, without regard to the type of decision or the test-specific marginal posterior summaries. This implies that, we assumed a uniform loss/reward function for the decisions. In the Bayesian context, we can formalize these ideas using the decision theoretic approaches. Under general settings,

$$L_0(.|d_i = 0) = \begin{cases} f_{00}(.) & \text{if Null true} \\ f_{01}(.) & \text{if Alternative true} \end{cases}$$

$$L_1(.|d_i = 1) = \begin{cases} f_{10}(.) & \text{if Null true} \\ f_{11}(.) & \text{if Alternative true} \end{cases}$$

where L_0 is the loss when the decision is to fail to reject the null and L_1 is the loss incurred when the null is rejected. The specification of the loss functions gives the flexibility in rewarding some genes differently than others. Typically, one assumes a uniform loss function in the absence of any specific information. That is, if we choose, $f_{00} = f_{01} = 0$ and $f_{10} = \lambda$, $f_{11} = -1$, we are essentially maximizing the true positives, while keeping the false negatives below certain value. The expected posterior loss in this case is given as:

$$-\sum_{i=1}^{I} d_i v_i + \lambda \sum_{i=1}^{I} d_i (1 - v_i)$$

where $v_i = E[\gamma_i]$. The optimal decision sequence d^* minimizes the expected posterior loss and λ shall be caliberated so that the estimated FDR is below the user-specified bound. Under the above settings, one can find optimal solution easily, given by:

$$\begin{aligned} &\max_\kappa s.t \quad && \frac{1}{\kappa} \sum_{j=1}^{\kappa} (1 - v_{(j)}) \leq \alpha \\ &\text{Reject} \quad && \kappa \text{ many genes with the largest } v_i's \end{aligned}$$

The solution can determined easily because, the odds ratio $\frac{v_i}{1-v_i}$ is monotonic in the weights. That is, if$v_i > v_{i'}$, then $\frac{v_i}{1-v_i} > \frac{v_{i'}}{1-v_{i'}}$. Instead, if one chooses gene dependent loss functions which reward true positives in a non-uniform fashion, then the posterior expected loss is given by:

$$-\sum_{i=1}^{I} d_i v_i f_i^* + \lambda \sum_{i=1}^{I} d_i (1 - v_i)$$

where $f_i^* = E_i(f_{11})$, is the expectation with respect to the posterior marginal distribution of the i-th gene. We can make the

dependence of the decisions on the false discovery rate (or some other measure) explicit by re-expressing the above objective function as:

$$\begin{aligned} \arg\max_{d*} \quad & \sum_{i=1}^{I} d_i v_i f_i^* \\ s.t \quad & \sum_{i=1}^{I} d_i(1-v_i) \leq \sum_{i=1}^{I} d_i \alpha \end{aligned}$$

where, $\sum_{i=1}^{I} d_i(1-v_i)$ is the expected false positives and $\sum_{i=1}^{I} d_i v_i$ is the expected true positives and α is the desired FDR. By the way, we still did not specify what f_i^* should be but just said that f_i^* is not a constant anymore. Consequently, we may loose monotonicity of the odds ratio, which in this case is, $\frac{v_i f_i^*}{1-v_i}$. As a result, the solution for the uniform loss function is no longer optimal. One could perform grid-based searches Mueller et al. (2004) or choose the thresholds Scott and Berger (2003). Our question is, can we obtain the optimal solutions in this case? What we mean by a solution is, finding the optimal decision sequence without resorting to bootstrapping or grid-searches or such approximation techniques. One could pretend that the odds ratio is monotonic and employ the global thresholding techniques which are greedy, but we only find a sub-optimal solution. In the next section, we briefly discuss the knapsack problem that has striking similarities to the problem at hand.

3. The 0-1 Knapsack problem

Consider again P genes with the i-th gene having a cost w_i (a positive integer) and profit v_i(a non-negative number). Let C (a non-negative integer) be the capacity of the knapsack. Our goal is to fill a knapsack with as many genes as possible maximizing profit but not fill the knapsack beyond its capacity, thus keeping the cost below a threshold. Stated formally, the objective is to Kellerer et al. (2004):

$$\begin{aligned} \arg\max_{d*} \quad & \sum_{i=1}^{I} d_i v_i \\ \text{s.t} \quad & \sum_{i=1}^{I} d_i w_i \leq C \end{aligned}$$

The formulation above is, in spirit, the same as α-investing in Foster and Stine (2008) and ideas in Peña et al (2010), who consider multiple hypothesis testing as a resource allocation problem. We want to maximize profits (true positives) while keeping the costs (false positives) down. These ideas have lead us to consider multiple hypothesis testing from an Operations Research (OR) perspective. Since our decision (or action) space is the collection of all P-tuples, it is essentially a combinatorial optimization problem Korte and Vygen (2008). For large P, obtaining the optimal solution by enumerating 2^P possible solutions is NP-hard and therefore is non-trivial. In the above problem, suppose that

$$\frac{v_1}{w_1} < \frac{v_2}{w_2} < ... < \frac{v_P}{w_P}.$$

That is, the profit/cost is an increasing function w.r.t the cost, then the optimal strategy to fill the knapsack is to simply pack all genes with the lowest cost first, until the capacity is reached . This is in fact the same solution we got in the previous section for the multiple hypothesis testing problem with uniform loss function, where we have monotonicity for the odds ratio. This version of the solution is called a greedy method in the OR literature. Even when the profit/cost does not have any specific pattern, the knapsack problem has some recurring substructures whose optimal solutions can obtained efficiently. Key features of the knapsack problem are:

- optimal substructure: an optimal solution to the problem contains within it optimal solutions to subproblem
- overlapping substructures: some subproblems will be visited again and again

More specifically, let `KP(i,c)` denote the optimal solution for the above problem. Then,

- If $d_P = 0$, that is if we do not place P-th gene in the knapsack, then, $d_1, d_2, \ldots, d_{P-1}$ must be the optimal solution for the problem `KP(P-1,c)`
- If $d_P = 1$, then, $d_1, d_2, \ldots, d_{P-1}$ must be the optimal solution for the problem `KP(P-1,c-`w_M`)`

More specifically,

$$\texttt{KP[i,c]} \quad = \quad \max(\texttt{KP[i-1,c]}, \texttt{KP[i-1,c-}w_i\texttt{]} + v_i)$$

The table `KP[,]` contains all the information to determine the optimal decisions for any given capacity not exceeding C. Pseudo code for the completing the table is given in the Appendix. Optimal state with the maximum profit for the given capacity is obtained by traversing the table in a specific manner presented in the Appendix. Suppose we set i=P and c=C-1 in Algorithm-2 given in the Appendix, we would get the optimal decision sequence with capacity bounded by C-1. Dynamic Programming (DP) principles help us to generate uniformly optimal sequential decisions for all capacities. In other words, if we we have the table `KP` computed for `KP(P,C-1)`, we only need to update the table by adding a column to it, without changing the first $C-1$ columns. This version of the knapsack is known as all-capacity knapsack and DP solves the all-capacity knapsack using the same resources (time and memory) as it takes for the knapsack with the largest capacity among them. This feature is particularly helpful to report decisions, profits and costs for a range of capacities. For a comprehensive review of knapsack problems, refer Kellerer et al. (2004), Korte and Vygen (2008). In the next section, we apply the above principles to the Bayesian multiple hypothesis testing.

4. Multiple hypothesis testing as a 0-1 Knapsack problem

For illustration purposes, consider the uniform loss case. By setting

$$\begin{aligned} v_i &= P(\gamma_i = 1) \\ w_i &= P(\gamma_i = 0) = 1 - v_i \end{aligned}$$

in the knapsack problem, we can obtain the optimal decision region but there is technical difficulty. First, we do not know $P(\gamma_i = 1)$, so we replace it by its posterior estimate, i.e.

$$\begin{aligned} v_i &= \hat{P}(\gamma_i = 1) \\ w_i &= \hat{P}(\gamma_i = 0) = 1 - v_i \end{aligned}$$

However, we require the costs to be positive integers and profits be non-negative. If we have genes with zero costs, the optimal strategy is to always pack them. Therefore, we will declare all genes with zero costs, irrespective of the profits, as differentially expressed. To apply the knapsack framework for the remaining genes, recognize that these estimated proportions ($\hat{P}(\gamma_i = 0)$) are rational numbers because we use B number of samples obtained using MCMC or some other inference engine. In other words, $\hat{P}(\gamma_i = 0) = \frac{1}{B}\sum_b \gamma_i^{(b)}$ and we actually have the costs specified as positive integers by construction:

Null true	Alternate true	
$B - X_i$	$X_i = \sum_b \gamma_i^{(b)}$	B

Therefore, if we set,

$$v_i = X_i, w_i = B - X_i$$

we can use DP to find the optimal decisions. False discovery rate can be estimated at the optimal decisions as:

$$\hat{\alpha} = \frac{1}{\sum_i d_i}\sum_i d_i(1 - v_i) = \frac{1}{B\sum_i d_i}\sum_i d_i w_i$$

It is nothing but the average cost of the genes in the knapsack scaled by the number of posterior samples used. We state the connection between Bayesian multiple hypothesis with generic loss functions and the 0-1 knapsack problem as follows:

Proposition:. *For any loss functions f_{01}, f_{10} with non-negative integer valued posterior expectations penalizing incorrect decisions and for any loss functions f_{00}, f_{11} with non-negative posterior expectations rewarding correct decisions, an optimal decision sequence can be obtained that maximizes the profts not the exceeding given loss bounded by C, using dynamic programming, with worst case complexity $O(CP)$.*

The requirements layed out in the above proposition are not as restrictive as they appear to be. One could simply set $f_{00} = 0$, $f_{01} = 0$, $f_{10} = 1$ and bring flexibility be altering $f_{11} > 0$. This allows one to control an aspect of the true positives but keeps a bound on the false positives. It is based on the same reasoning for constraining the costs and profits in the knapsack problem to be non-negative. For example, if both are negative, we can create consider them as positive but invert the decisions. These technical requirements ensure that the algorithm is simple to manage and maintain. It does not take away flexibility in designing sensible loss functions. By no means, the framework is restricted to controlling FDR. For example, consider $f_{00} = -\lambda$, $f_{01} = 1$, $f_{10} = 0$, $f_{11} = 0$, then we would be maximizing true negatives while keeping a bound on the false negatives. A complete algorithm in the form of pseudo code is given the Appendix which reports the optimal decision for a given FDR.

5. Simulation examples and Discussion

Example-1: Let us begin with a version of the simple example considered in pp. 16, Kellerer et al. (2004). We have seven genes and the knapsack's capacity is 0.9. Costs and profits of the genes are given below:

i	1	2	3	4	5	6	7
v_i	0.6	0.5	0.3	0.6	0.8	0.9	0.7
w_i	0.2	0.3	0.4	0.5	0.6	0.7	0.9
$\frac{v_i}{w_i}$	3	1.67	0.75	1.2	1.33	1.29	0.78

When using the greedy approach, we pick the gene with largest profit/cost first and repeat this until the knapsack reaches its capacity or we exhaust the genes. If a gene does not fit into knapsack, we simply skip and go to the next gene. So, we pick the 1st, 2nd and the 4th genes that exactly fill the knapsack. Using DP, we pick the 1st and the 4th genes. For the given capacity, the optimal solution had total value 15, while the greedy had 14. The greedy solution would have been the optimal solution, had the weights been ordered according to profit/cost.

Example-2: We simulate P=100 genes. To avoid any model specific assumptions and inferential goals, we generate γ_i i.i.d Beta(3, 1) with mean 0.75. Then, for each gene i, we generate B=100 Bernouli random variables with success probability γ_i, i.e., $\gamma_i^{(b)}$ i.i.d Bin$(1, \gamma_i)$. The costs assigned to the genes in the knapsack are $w_i = B - X_i = B - \sum_b \gamma_i^{(b)}$ with corresponding profits $v_i = \frac{X_i}{B}$. In this case, the greedy solution and the dynamic approach should produce identical results because the odds ratio or profit/cost is monotonic as shown in Fig. 1 (solid). In Fig. 2, we plot FDR vs the number of discoveries made (total number genes in the knapsack). Both DP and greedy algorithms produce identical results. Estimated FDR is plotted against the estimated TDR (true discovery rate, which is the average profit of genes in the knapsack) in Fig. 3. It is a decreasing function of FDR because as we increase the capacity, we add more genes having smaller profits which bring down the average profit. If profit/cost is monotonically decreasing with cost, then we expect TDR to drop down monotonically as well. Total capacity vs total profit of the genes in the knapsack is shown in Fig. 4 and as expected profit increases with capacity. We see from Fig. 5 that number of genes in the knapsack increase as a the capacity is increased.

Example-3: To simulate a more complex scenario that could be applicable with generic loss functions, we break the monotonicity of profit/cost by perturbing the profits randomly. The new profits are generated by multiplying the previously assigned profits by random weights drawn from U(0,1), i.e.,

$v_i^* = u_i * v_i, u_i \sim \text{U}(0,1)$. The resulting profit/cost ratio is shown Fig. 1 (dotted). We again run the greedy algorithm and DP algorithms with v_i^* and w_is. In Fig. 6, FDR vs the number of discoveries is plotted. Both DP and Greedy algorithms produce similar results. But looking at the estimated FDR vs the estimated TDR plot in Fig. 7, it is clear that greedy algorithm has lower TDR for a given FDR than the DP solution. That is, the DP knapsack has higher average profit than the greedy knapsack. It is worth noting that, TDR increases in the beginning with FDR and then falls-off. This is a consequence of breaking the profit/cost ordering. In Fig. 8, we plot the estimated power (average profit of genes in the knapsack) as a function of the estimated FDR which exhibits similar to behavior to Example-2. Both the simulation examples show some that DP framework can be used to solve multiple hypothesis testing problems under different scenarios.

Discussion: An important component of these combinatorial optimization problems is the time and space complexity. The 0-1 knapsack problem is a psedu-polynomial time algorithm in the sense that the complexity depends on capacity and it not bounded asymptotically. In the multiple hypothesis testing framework, this translates to how many posterior are samples needed to compute the optimal solution in a reasonable amount of time. A conservative estimate for B can be $\left\lceil \sqrt{\frac{1}{4e}} \right\rceil$, where e is the monte-carlo standard error for estimating the posterior probabilities $P(\gamma_i)$ assuming i.i.d samples. A loose bound for the capacity can now be given as $C \leq P \sum X_i$ (note that $X_i \leq B$) and the worst case complexity for the computing the table KP is $O(P^2 B)$. Often in practice, the desired FDR will be attained at a much lower capacity and finding tighter bounds a topic for future research Martello et al. (1999). We point that computing the table has lower complexity than determining the optimal decision sequence. This makes computing the FDR at every capacity, which is based optimal decision sequence at that capacity, can be impractical. However, it is possible to reduce this time by employing a leap-and-bound strategy (not implemented). That is, we estimate the FDRs at certain increments of the capacity and check if the FDR has exceeded. While we are not recommending any particular strategies, there are many choices for customizing the algorithm for the problem at hand and this is an active area of research on it own Martello et al. (2000). Several hard to solve knapsack problems are discussed in Psinger (2005).

6. Conclusion

We considered multiple hypothesis in the Bayesian context without making any specific assumptions about the underlying probability model. Assuming that there is an efficient mechanism to generate posterior samples for the indicator variables for the individual tests, we showed how the knapsack problem, a combinatorial optimization framework, can be adopted for the problem at hand. We provided an algorithm for maximizing true discoveries while keeping the FDR below the bound. Our approach solves the problem in the cohesive decision theoretic setup. We believe that discrete optimization is a feasible solution in a variety of multiple hypothesis testing scenarios, which can be solved exactly. We highlighted several features of the knapsack framework that could lend insights into the multiple hypothesis testing problem. In the event that capacity bounds are enormously large, we can leverage several approximate algorithms available in the knapsack problem literature. We hope that our contribution stimulates research to find better solutions.

Acknowledgements

The authors would like to thank Prof. Edsel Peña and Prof. Bob Stine for the inspiring talks presented during the Resampling Conference held at Texas A & M University. After-presentation discussion with Prof. Peña has lead us to pursue the multiple testing as a resource allocation problem.

Appendix A. Algorithms for the the 0-1 Knapsack problem

Algorithm 1: Computing the table

```
for c = 0 to C, K[0,c] = 0

for i = 0 to I, K[i,0] = 0

for i = 1 to I

    for c = 1 to C

        if w_i > c
           K[i,c]=K[i-1,c]
        else
           if v_i+K[i-1 ,c-w_i] > K[i-1,c],
           K[i,c] = v_i + K[i-1,c-w_i]
           else K[i,c] = K[i-1,c]
        end

    end

end
```

Algorithm 2: Finding items in the knapsack

```
set i=I,c=C.

do untill i=0

    if K[i,c] ≠ K[i-1,c]
        mark the i-th item as in the
        knapsack
        i=i-1, c= c-w_i
    else
        i=i-1
    end

end
```

Algorithm 3: FDR control

```
for c = 0 to C, K[0,c] = 0

for i = 0 to I, K[i,0] = 0

set c = 1, fdr[0] = 0

do until fdr[c] > α or c = C+1

    for i = 1 to I
        if w_i > c
            K[i,c]=K[i-1,c]
        else
            if v_i+K[i-1 ,c-w_i] > K[i-1,c],
            K[i,c] = v_i + K[i-1,c-w_i]
            else K[i,c] = K[i-1,c]
        end
    set j = c, d_i = 0∀i
    do until i=0
        if K[i,j] ≠ K[i-1,j]
            set d_i = 1
            i=i-1, j= j-w_i
        else
            i=i-1
        end
    end
    fdr[c] = (B Σ_i d_i)^-1 Σ_i d_i w_i
    M[c,i] = d_i∀i
    c = c+1

end

report fdr[c-1] and the decisions d_i^* = M[c − 1, i] ∀i
```

References

Benjamini, Y. and Hochberg, Y. (1995). Controlling the false discovery rate: a practical and powerful approach to multiple testing. *J. Roy. Statist. Soc. Ser. B* **57**, 1, pp.289-300

Bernhard H. Korte, Jens Vygen (2008). Combinatorial optimization: theory and algorithms. *Springer*

Chuanwen Chen, Arthur Cohen, and Harold B. Sackrowitz (2009). Admissible, consistent multiple testing with applications including variable selection. *Electron. J. Statist.* **3**, pp.633-650

Efron, B. (2008). Simultaneous inference: When should hypothesis testing prob- lems be combined? *The Annals of Applied Statistics* **1**, pp.197-223.

Ferkingstad, E., Frigessi, A., Rue, H., Thorleifsson, G., and Kong, A. (2008). Unsupervised empirical Bayesian multiple testing with external covariates. *The Annals of Applied Statistics* **2**, pp.714-735

Foster, D. P. and Stine, R. A. (2008). α-investing: a procedure for sequential control of expected false discoveries. *J. R. Stat. Soc. Ser. B Stat. Methodol. 70*, 2, pp. 429-444.

Genovese, C. and Wasserman, L. (2002). Operating characteristic and extensions of the false discovery rate procedure. *Journal of the Royal Statistical Society, Series B* **64**, pp.499-517.

Gottardo, Raphael, Raftery, E. Adrian, Y. Yeung, Ka, Bumgarner, and E. Roger (2006). Bayesian Robust Inference for Differential Gene Expression in Microarrays with Multiple Samples, *Biometrics*, **62**, (1), pp. 10-18

Guindani, M., Zhang, S. and Mueller, P.M. (2009). A Bayesian discovery procedure. *JRSS- B*, **71**(5). pp.905-925.

Kellerer, H., Pferschy, U., and Pisinger, D. (2004). Knapsack problems. *Springer*

Kim, S., Dahl, D. B., Vannucci, M. (2009), Spike and Slab Dirichlet Process Prior for Multiple Hypothesis Testing in Random Effects Models, *Bayesian Analysis*, **4**, pp.707-732

Martello, S., Pisinger, D. and Toth, P. (1999). Dynamic programming and strong bounds for the 0-1 knapsack problem. *Management Science*, **45**(3) pp. 414424

Martello, S., Pisinger, D. and Toth, P. (2000) New trends in exact algorithms for the 0-1 knapsack problem. *European Journal of Operational Research*, **123** pp.325332

Muller, P., Parmigiani, G., and Rice, K. (2007) Fdr and bayesian multiple comparisons rules. In *Bayesian Statistics 8* (eds. J. Bernardo, M. Bayarri, J. Berger, A. Dawid, Heckerman, A. D., Smith and M.West). Oxford, UK: Oxford University Press

Muller, P., Parmigiani, G., Robert, C. P, and Rousseau, J. (2004) Optimal sample size for multiple testing: the case of gene expression microarrays. *Journal of the American Statistical Association*, **99**, pp.9991001.

Peña A. E., E. A, Habiger, J. D. and Wu, W. (2010), Power-enhanced multiple decision functions controlling family-wise error and false discovery rates. *submitted: Annals of Statistics*

Pisinger, D *Where are the hard knapsack problems? Computers & Operations Research*, **32**(9) pp. 22712284 codes `http://www.diku.dk/hjemmesider/ansatte/pisinger/codes.html`

Roquain, E. and van de Wiel, M. A. (2009). Optimal weighting for false discovery rate control. *Electronic Journal of Statistics* **3**, pp.678-711.

Sarkar, S. K., Zhou, T., and Ghosh, D. (2008). A general decision theoretic formulation of procedures controlling FDR and FNR from a Bayesian perspective. *Statist. Sinica* **18**, 3, pp.925-945.

Scharpf, R.B, Tjelmeland, H., Parmigiani, G., and Nobel, A. (2009). A Bayesian model for cross-study differential gene expression. *Journal of the American Statistical Association*, **104** (488)

Scott, J. and Berger, J. (2003) An exploration of aspects of Bayesian multiple testing. *Journal of Statistical Planning and Inference*, **136**, pp.21442162.

Storey, J. (2003). The positive false discovery rate: a Bayesian interpretation and the q-value. *The Annals of Statistics* **31**, pp.2012-2035

Storey, J (2007) The optimal discovery procedure: a new approach to simultaneous significance testing. *J. R. Statist. Soc. B*, **69** (3), pp.347368.

Storey, J., Dai, J. and Leek, J. (2007) The optimal discovery procedure for large-scale significance testing, with applications to comparative microarray experiments. *Biostatistics*, **8**, pp.414432.

Sun, W. and Cai, T. (2007). Oracle and adaptive compound decision rules for false discovery rate control. *Journal of the American Statistical Association* **102**, pp.901-912.

Tang, W. and Zhang, C-H. (2007), Empirical Bayes Methods for Controlling the False Discovery Rate with Dependent Data it Lecture Notes-Monograph Series, Vol. 54, Complex Datasets and Inverse Problems: Tomography, Networks and Beyond,pp. 151-160

Westfall, P. and Troendle, J. (2008). Multiple testing with minimal assumptions. *Biometrical Journal* **50**, pp.1-11.

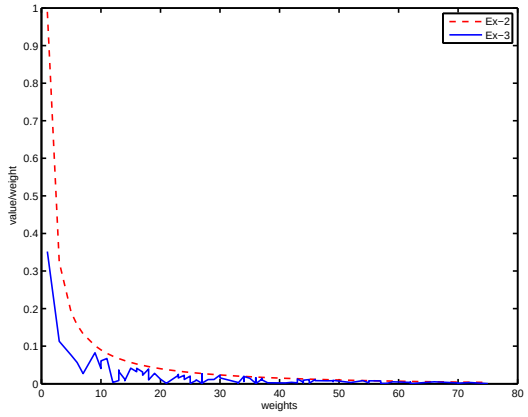

Figure A.1: Profit/cost in Example-2 (solid), Example-3(dashed)

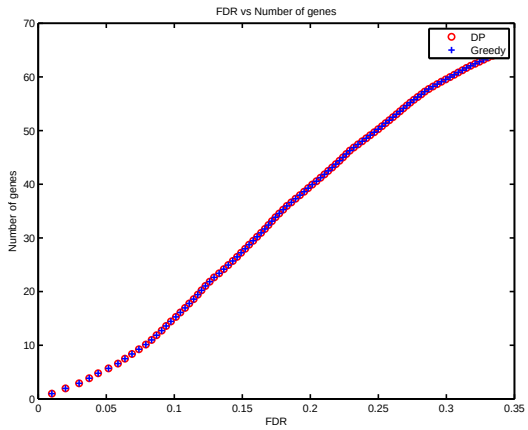

Figure A.2: False Discovery Rate vs Number of discoveries for genes in Example-2 using DP(o) and Greedy(+) algorithms

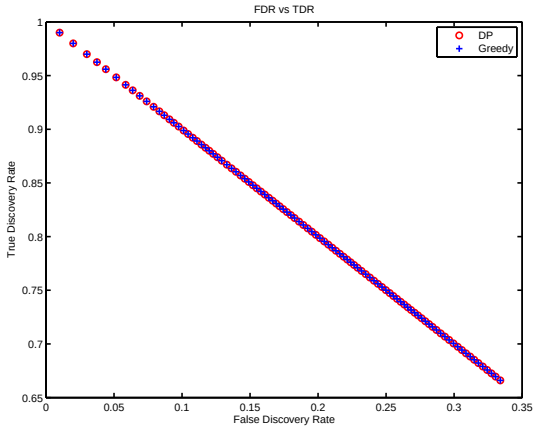

Figure A.3: FDR vs TDR for genes in the knapsack for Example-2 using DP(o) and Greedy(+) algorithms

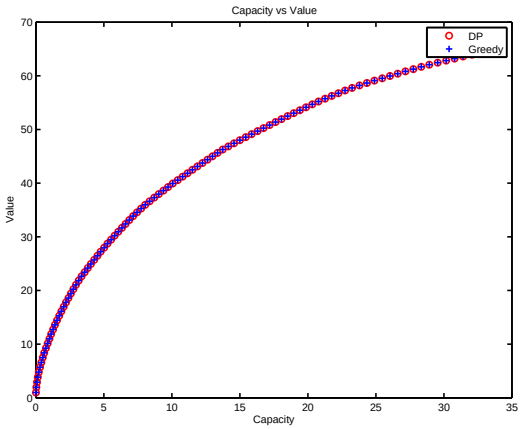

Figure A.4: Total cost vs Total profit of genes in the knapsack for Example-2 using DP(o) and Greedy(+) algorithms

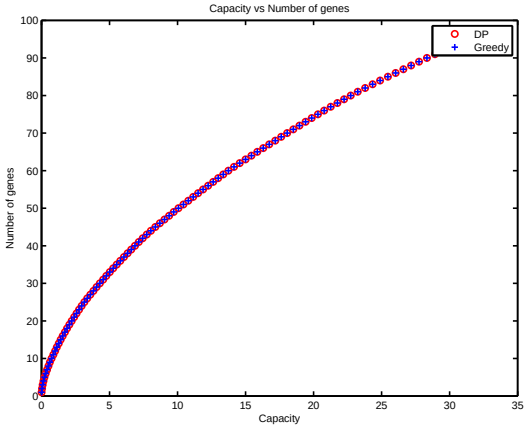

Figure A.5: Total capacity vs Number of genes in the knapsack for Example-2 using DP(o) and Greedy(+) algorithms

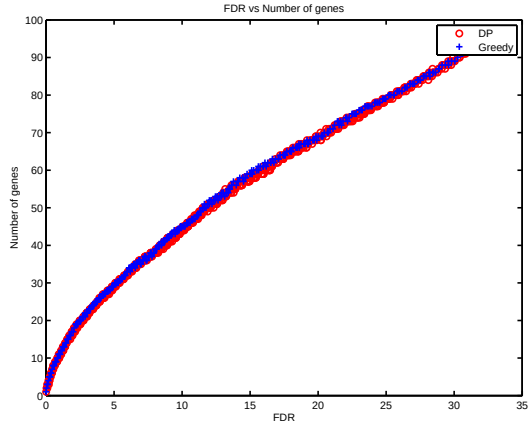

Figure A.6: False Discovery Rate vs Number of discoveries for genes in Example-3 using DP(o) and Greedy(+) algorithms

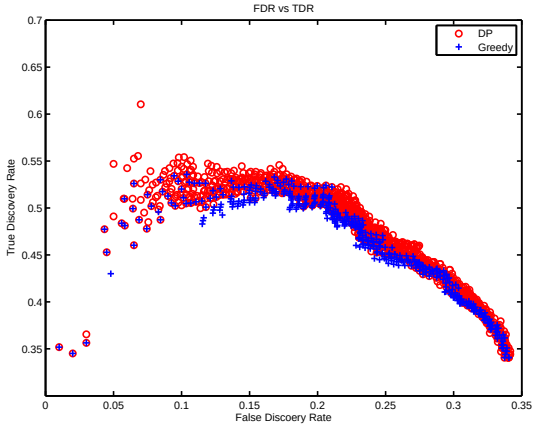

Figure A.7: False Discovery Rate vs TDR for genes in Example-3 using DP(o) and Greedy(+) algorithms

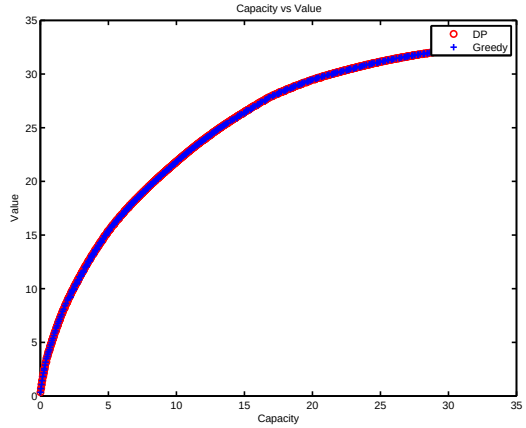

Figure A.8: Total cost vs Total profit of genes in the knapsack for Example-3 using DP(o) and Greedy(+) algorithms