Dirichlet Distribution (courtesy of Prakash Laud)

$$(x_1, \cdots, x_P) \sim D(\alpha_1, \cdots, \alpha_P)$$

$$x \sim D(\alpha)$$

$$[x_1, \cdots, x_{P-1}] = \frac{\Gamma(\alpha_0)}{\prod_{j=1}^{P} \Gamma(\alpha_j)} \left\{ \prod_{j=1}^{P} x_j^{\alpha_j - 1} \right\}$$

$$\text{where } 0 \leq x_j \leq 1,\ x_P = 1 - \sum_{j=1}^{P-1} x_j,\ \alpha_j > 0,\ \alpha_0 = \sum_{j=1}^{P} \alpha_j$$

$$E(x_j) = \frac{\alpha_j}{\alpha_0}$$

$$Var(x_j) = \frac{\alpha_j(\alpha_0 - \alpha_j)}{\alpha_0^2(\alpha_0 + 1)}$$

$$Cov(x_j, x_k) = \frac{-\alpha_j \alpha_k}{\alpha_0^2(\alpha_0 + 1)}$$

Dirichlet Properties (courtesy of Prakash Laud)

- **Univariate marginals**
  $x_j \sim Beta(\alpha_j, \alpha_0 - \alpha_j)$
- **Multivariate marginals by rescaling**
  $\left(\frac{x_1}{\sum_{j=1}^{Q} x_j}, \cdots, \frac{x_Q}{\sum_{j=1}^{Q} x_j}\right) \sim D(\alpha_1, \cdots, \alpha_Q)$ where $2 \leq Q \leq P$
- **Collapsed cells property**
  $(x_1 + x_2, x_3, \cdots, x_P) \sim D(\alpha_1 + \alpha_2, \alpha_3, \cdots, \alpha_P)$
- **Conditional distributions by rescaling**
  $\left(\frac{x_1}{1-\sum_{j=Q+1}^{P} x_j}, \cdots, \frac{x_Q}{1-\sum_{j=Q+1}^{P} x_j} | x_{Q+1}, \cdots, x_P\right) \sim D(\alpha_1, \cdots, \alpha_Q)$
- **Conjugacy with Multinomial**

$$x | \theta \sim M(n, \theta), \quad \theta \sim D(\alpha) \Rightarrow \theta \mid x \sim D(\alpha + x)$$

The DART prior: BART with sparse variable selection

- Alternatively, for variable selection, you can specify a Dirichlet prior which is more appropriate if the number of covariates is large (Linero 2018, JASA)
- we can represent the probability via the sparse Dirichlet prior as $[s_1, ..., s_P] \,|\theta \overset{\text{prior}}{\sim} \mathbf{D}\,(\theta/P, ..., \theta/P)$ which is specified by the argument `sparse=TRUE` while the default is `sparse=FALSE` for uniform $s_j = P^{-1}$
- The prior parameter θ can be fixed or random: supplying a positive number will specify θ fixed at that value while the default `theta=0`, set to zero, specifies a random value learned from the data
- The random θ prior is induced via $\theta/(\theta + \rho) \overset{\text{prior}}{\sim} \mathbf{Beta}\,(a,\ b)$ where the parameter ρ can be specified by the argument `rho` (which defaults to 0, zero, representing the value P; provide a value to over-ride), the parameter b defaults to 1 (which can be over-ridden by the argument `b`) and the parameter a defaults to 0.5 (which can be over-ridden by the argument `a`)

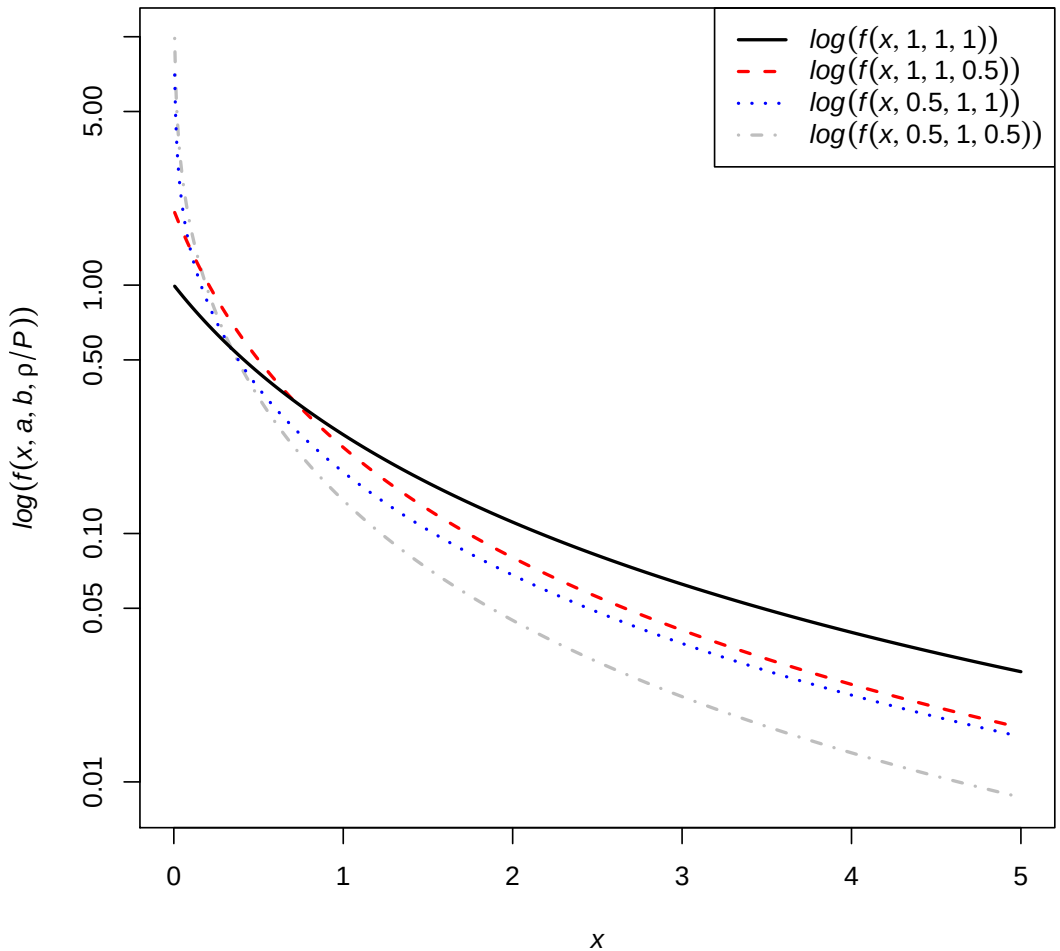
The DART prior

- The distribution of `theta` controls the sparsity of the model: `a=0.5` induces a sparse posture while `a=1` is not sparse and similar to the uniform prior with probability $s_j = P^{-1}$
- If additional sparsity is desired, then you can set the argument `rho` to a value smaller than P
- Here, we take the opportunity to provide some insight into how and why the sparse prior works as desired
- The key to understanding the inducement of sparsity is the distribution of the arguments to the Dirichlet prior: θ/P
- it can be shown that $\theta/P \sim F(a, b, \rho/P)$ where $F(.)$ is the Beta Prime distribution scaled by ρ/P
- The non-sparse setting is $(a, b, \rho/P) = (1, 1, 1)$
- As we will see, sparsity is increased by reducing ρ : $(1, 1, 0.5)$; reducing a : $(0.5, 1, 1)$ which is the default; and even moreso by reducing both: $(0.5, 0.5, 1)$

The DART prior

The distribution of $\boldsymbol{\theta/P}$ and the sparse Dirichlet prior

Sparapani, Spanbauer and McCulloch 2021 *JSS*



Posterior computation for DART

- Posterior computation related to the Dirichlet sparse prior
- If a Dirichlet prior is placed on the variable splitting probabilities, s, then its posterior samples are drawn via Gibbs sampling with conjugate Dirichlet draws
- The Dirichlet parameter is updated by adding the total variable branch count over the ensemble, m_j, to the prior setting, $\frac{\theta}{P}$, i.e., $\left[\frac{\theta}{P} + m_1, ..., \frac{\theta}{P} + m_P\right]$ (Multinomial conjugacy)
- In this way, the Dirichlet prior induces a "rich get richer" variable selection strategy
- The sparsity parameter, θ, is drawn on a grid of values
- This draw only depends on $[s_1, ..., s_P]$
- BART R package: each variable's branch count is returned in the fit object: `varcount` and `varcount.mean`
- And the probabilities are returned too: `varprob` and `varprob.mean`

DART with grouped variables

Chipman, George et al. 2021; Chapter: Computational approaches to Bayesian Additive Regression Trees; Book: Computational Statistics in Data Science

- We have P variables, but Q of them encode a grouped variable such as dummy indicators for a categorical variable (these are the first Q variables without loss of generality): $x_1, ..., x_Q$
- N.B. This applies to multiple grouped variables; however, for brevity, a single grouped variable will suffice
- The variable selection probabilities are $s = [s_1, ..., s_P]$
- There are two other probability collections of interest
- The collapsed probabilities, $p = [s_1 + \cdots + s_Q, s_{Q+1}, ..., s_P]$
- And the re-scaled probabilities $q = [\tilde{s}_1, ..., \tilde{s}_Q]$ where $\tilde{s}_j \propto s_j$ such that $\sum_{j=1}^{Q} \tilde{s}_j = 1$

DART with grouped variables

- Blindly using Dirichlet variable selection probabilities, then we arrive at the following
- $s|\theta \overset{\text{prior}}{\sim} \mathrm{D}_{P}(\theta/P, ..., \theta/P)$ where the subscript P is the order of the Dirichlet
- $p|\theta \overset{\text{prior}}{\sim} \mathrm{D}_{\widetilde{P}}(Q\theta/P, \theta/P, ..., \theta/P)$ where $\widetilde{P} = P - Q + 1$
- $q|\theta \overset{\text{prior}}{\sim} \mathrm{D}_{Q}(\theta/P, ..., \theta/P)$
- The problem: the distribution of p_1, the first element of p, puts more prior weight on the grouped variable than the others

DART with grouped variables

- The solution to the problem is trivial: re-scale $\boldsymbol{q}$ by $\boldsymbol{Q}^{-1}$ while naturally re-defining $\boldsymbol{p}$ and $\boldsymbol{s}$ as follows.

$$
\begin{aligned}
\boldsymbol{p}|\boldsymbol{\theta} &\overset{\text{prior}}{\sim} \mathrm{D}_{\widetilde{\boldsymbol{P}}}\left(\boldsymbol{\theta}/\widetilde{\boldsymbol{P}}, ..., \boldsymbol{\theta}/\widetilde{\boldsymbol{P}}\right) \\
\boldsymbol{q}|\boldsymbol{\theta} &\overset{\text{prior}}{\sim} \mathrm{D}_{\boldsymbol{Q}}\left(\boldsymbol{Q}^{-1}\boldsymbol{\theta}/\widetilde{\boldsymbol{P}}, ..., \boldsymbol{Q}^{-1}\boldsymbol{\theta}/\widetilde{\boldsymbol{P}}\right) \\
\boldsymbol{s}|\boldsymbol{\theta} &\overset{\text{prior}}{\sim} \mathrm{D}_{\boldsymbol{P}}\left(\boldsymbol{Q}^{-1}\boldsymbol{\theta}/\widetilde{\boldsymbol{P}}, ..., \boldsymbol{Q}^{-1}\boldsymbol{\theta}/\widetilde{\boldsymbol{P}}, \boldsymbol{\theta}/\widetilde{\boldsymbol{P}}, ..., \boldsymbol{\theta}/\widetilde{\boldsymbol{P}}\right) \\
&\overset{\text{prior}}{\sim} \mathrm{D}_{\boldsymbol{P}}\left((\boldsymbol{q}|\boldsymbol{\theta}), (\boldsymbol{p}|\boldsymbol{\theta})\right)
\end{aligned}
$$

- The BART3 R package's `gbart` function takes this approach automatically when you supply a data frame with the covariates where the categorical variables are factors (rather than supplying a matrix for the covariates)

Thompson Sampling Variable Selection (TSVS)

Liu and Rockova, JASA 2023

- A stochastic optimization approach to subset selection based on reinforcement learning as an extension of the DART model based on Thompson Sampling (Russo, Van Roy et al. 2018 *Foundations and Trends in Machine Learning*)
- It can be regarded as a multi-armed bandit problem where each variable is treated as an arm
- $y_i = f(x_i) + \epsilon_i$ where $\epsilon_i \overset{iid}{\sim} N(0, \sigma^2)$
- Variable selection problem: determine the optimal subset, or near-optimal, $S_O \subset \{1, ..., P\}$ with cardinality $Q_O = |S_O|$ predictors that have an impact on the fit $f(x_i)$
- The variable inclusion probabilities have Beta priors $\theta_j \overset{\text{ind}}{\sim} \text{Beta}(a_j, b_j)$ where $j = 1, ..., P$
- γ_j: an unknown Bernoulli reward if x_j is chosen $\gamma_j \overset{\text{ind}}{\sim} \text{B}(\theta_j)$
- θ_j: the unknown mean reward is the inclusion probability $\text{E}[\gamma_j] = P(\gamma_j = 1 | \theta_j) = \theta_j$

Multi-armed Bandits (MAB)

- MAB: Decide which of P arms to play at step t, given the outcome of the previous $t-1$ steps where $t = 1, ..., T$
- Goal: maximize sum of expected rewards and minimize regret
- Multi-play Scenario: At each step t, select a subset S_t of arms and receive binary rewards of all selected arms
- Reward, $\gamma_j(t)$: $\gamma_j(t) \overset{\text{ind}}{\sim} B(\theta_j(t))$
  N.B. this is Liu's notation: typically, it would be γ_{jt}, θ_{jt}
- Maximize the sum of expected rewards over the drawn arms
- Optimal action: select arms $S_O(t) = \{j : \gamma_j(t) = 1\}$
- Regret, $\mathcal{R}(T)$: expected cumulative reward difference between the optimal drawing policy and the selected draws

$$E[\mathcal{R}(T)] = E\left\{ \sum_{t=1}^{T} \left(\sum_{j \in S_O} \theta_j(t) - \sum_{j \in S_t} \theta_j(t) \right) \right\}$$

Multi-armed Bandits (MAB)

- Global Reward, $R_C(S)$: a computational oracle
  regret minimizer when an oracle furnishes probabilities $\theta_j(t)$

$$R_C(S_t) = \sum_{i \in S_t} \log(C + \gamma_j(t))$$

$$r_\theta^C(S_t) = E[R_C(S_t)] = \sum_{i \in S_t} \left[\theta_j(t) \log\left(\frac{C+1}{C}\right) - \log\left(\frac{1}{C}\right) \right]$$

- Computational Oracle, S_O: $S_O = \arg\max_S r_\theta^C(S)$

$$S_O = \left\{ j : \theta_j(t) \geq \frac{\log(1/C)}{\log[1 + 1/C])} \right\}$$

Setting $C = (\sqrt{5} - 1)/2$ gives the median probability model

$$S_O = \{ j : \theta_j(t) \geq 0.5 \}$$

TSVS Algorithm for High Dimensions: Big P or Big N

Initialize parameters: you may need to experiment with those in red to get adequate performance especially M and T

- $\tilde{C} = \frac{\log(1/C)}{\log(1+C)/C}$ for some $0 < C < 1$ (typically, $\tilde{C} = 0.5$)
- L, length of DART chain burn-in discarded
- M, length of DART chain to keep
  N.B. typically, you have to run DART serially, i.e., NOT with parallel processing since the effective lengths of the chain in parallel would be $M/$`mc.cores` rather than M
- H, number of trees: typically, $H = 10$
- T, number of steps
- $a_j(0) = a > 0, \quad b_j(0) = b > 0 \quad \text{where } j = 1, ..., P$

TSVS Algorithm

For $t = 1, ..., T$

a. For $j = 1, ..., P$, draw $\theta_j(t) \sim Beta(a_j(t-1), b_j(t-1))$

b. Set $S_t = \{j : \theta_j(t) \geq \tilde{C}\}$

c. Fit DART model $f_t(x(t))$ with $x_j(t)$ **where** $j \in S_t$

d. For $j = 1, ..., P$

(i) If $j \notin S_t$, then set $\gamma_j(t) = 0$
Else calculate reward $\gamma_j(t)$ from DART fit $f_t(.)$

(ii) Set $a_j(t) = a_j(t-1) + \gamma_j(t)$

(iii) Set $b_j(t) = b_j(t-1) + 1 - \gamma_j(t)$

(iv) Calculate inclusion probability $\pi_j(t) = \frac{a_j(t)}{a_j(t)+b_j(t)}$

Trajectories of important covariates for $\pi_j(t)$ will exceed 0.5 by T

TSVS Algorithm: "Offline" for Big P

- N.B. there are no limits on P
- For example, TSVS can be used when $P >> N$
- Typically, $M = 1000$
- If $j \in S_t$, then set $\gamma_j(t) = 1$ when the corresponding `varcount` for the Mth draw is $m_{jM} > 0$
- Otherwise, set $\gamma_j(t) = 0$
- Liu and Rockova recommend $T = 500$, but our experience has been that $T = 20$ or 50 is often all that is needed

TSVS Algorithm: "Online" Big $N >> P$ with sharding

- Typically, $M = 10000$
- If $j \in S_t$, then set $\gamma_j(t) = 1$ when the corresponding `varcount.mean` for the M draws is $M^{-1} \sum_k m_{jk} = \overline{m}_j \geq 1$
- Otherwise, set $\gamma_j(t) = 0$
- Typically, $T = 100$
- The data set is partitioned into shards of size N/T and at each step you progress through the shards rather than the whole data set which is too big for DART to process efficiently
- However, due to the performance of TSVS, you may need to pass through the data set multiple times with bootstrapping
- So, you might consider B bootstrap passes through the data $T = A \times B$ with random shards of size N/B
- Typically, $B = 5$ and $A = 20$

Diabetes and recurrent hospital admissions

- A cohort of newly diagnosed diabetes patients and their hospital admissions (and occasionally multiple admissions) from a single health care system
- We have the electronic health records (EHR) for these patients from 2007-2012: prior records may, or may not, be available
- EHR are an omnibus of digital health care information
- We focus on 84 covariates: time, number of previous admissions, patient demographics, health insurance, health care charges, diagnoses, procedures, anti-diabetic therapy, laboratory values and vital signs
- By its nature, EHR data is fundamentally time-varying
- EHR covariates are occasionally missing at time zero even when carrying the last value forward so we imputed 15 continuous variables with Sequential BART (Xu, Daniels & Winterstein 2016 *Biostatistics*)

Diabetes and recurrent hospital admissions

- 488 patients followed 5 years from 2008-2012
  the survival rate was high 458/488=0.939
  and hospital admissions were more than one apiece: 525 total
- For diabetes, which covariates increase the risk of admission? What about the number of previous admissions or an acutely recent admission?
- What are the functional forms of the covariates, e.g., linear, quadratic, logarithm, etc.? Are the covariate effects additive or multiplicative?
- Are there interactions?
- We want to avoid precarious restrictive assumptions hence we choose to do variable selection with BART

Diabetes and recurrent hospital admissions

	Patients		Admissions	
Number of Admissions	488		525	
0	308	(63.0)	0	
1	79	(16.2)	79	(15.0)
2-3	50	(10.3)	115	(21.9)
4-16	51	(10.5)	331	(63.1)

Diabetes and recurrent hospital admissions

	Patients		Admissions	
Gender	488		525	
M	216	(44.3)	228	(43.4)
F	272	(55.7)	297	(56.6)
Race	488		525	
Black	174	(35.7)	265	(50.5)
White	314	(64.3)	260	(49.5)
Age	488		525	
Mean, SD	60.9	15.0	60.3	15.7
ZIP3 area	488		525	
urban	378	(77.5)	454	(86.5)
suburb	110	(22.5)	71	(13.5)
Insurance and Age	488		525	
Government 65+	191	(39.1)	224	(42.7)
Government <65	138	(28.3)	208	(39.6)
Commercial <65	143	(29.3)	71	(13.5)
Other <65	16	(3.3)	22	(4.2)

Diabetes and recurrent hospital admissions

- We focus on 84 covariates: time, number of previous admissions, patient demographics, health insurance, health care charges, diagnoses, procedures, anti-diabetic therapy, laboratory values and vital signs
- Randomly divided into training and validation sets
- Training set: "fit the fit" (DSS) vs. TSVS
- DSS: Serum calcium, peripheral vascular disease (PVD), the number of previous admissions, $N_i(t-)$, insulin treatment, and peptic ulcer disease (PUD)
- TSVS: Serum calcium, peripheral vascular disease (PVD), the number of previous admissions, $N_i(t-)$, blindness, cardiomyopathy, creatinine, gangrene and Relative Value Units (RVU) 31 to 90 days earlier
  see `demo/diabetes.R` in the **BART3** package

TSVS variable selection plot