A REVIEW OF THE BRISTOL-MYERS SQUIBB / ASTRAZENECA ECONOMIC MODEL ON THE COST-EFFECTIVENESS OF DAPAGLIFLOZIN

REPORT BY THE DECISION SUPPORT UNIT

29th November 2012

Sarah Davis
School of Health and Related Research, University of Sheffield
Jefferson Sheard
Jefferson Sheard Consultancy, Sheffield

Decision Support Unit, ScHARR, University of Sheffield, Regent Court, 30 Regent Street
Sheffield, S1 4DA

Tel (+44) (0)114 222 0734
E-mail dsuadmin@sheffield.ac.uk
Website www.nicedsu.org.uk

ABOUT THE DECISION SUPPORT UNIT

The Decision Support Unit (DSU) is a collaboration between the Universities of Sheffield, York and Leicester. We also have members at the University of Bristol, London School of Hygiene and Tropical Medicine and Brunel University. The DSU is commissioned by The National Institute for Health and Clinical Excellence (NICE) to provide a research and training resource to support the Institute's Technology Appraisal Programme. Please see our website for further information www.nicedsu.org.uk

The production of this document was funded by NICE through its Decision Support Unit. The views, and any errors or omissions, expressed in this document are of the authors only. NICE may take account of part or all of this document if it considers it appropriate, but it is not bound to do so.

Acknowledgements

We thank Eldon Spackman for providing peer review comments and Jenny Dunn for formatting the report. The content of this report is the responsibility of the authors alone.

EXECUTIVE SUMMARY

The economic model submitted by Bristol-Myers Squibb / AstraZeneca is a patient level simulation. The model has an Excel front-end, but it also utilises C++ programming compiled into Dynamic Link Library (DLL) format. The DLL components of the economic model are bespoke pieces of software and are therefore not considered to fall within the list of standard software packages usually accepted by NICE.

The Decision Support Unit (DSU) has examined the submitted model, including the source code used to generate the DLLs, to determine whether the economic model functions as described in the submission and whether it produces the results reported in the submission. Whilst the DLLs are "black boxes" of code that cannot be investigated from the Excel model, it is possible to recompile the C++ source code and watch how it works in a similar way to validating visual basic code in Excel.

The DSU found several differences between the description of the model provided by the manufacturer and the executable model submitted. The main differences are as follows;

- There are some discrepancies between the event equations and risk factor equations implemented within the model and those described by the manufacturer.
- Treatment related weight changes are applied immediately in the model rather than being achieved gradually over the first year of treatment.
- All cause mortality is not adjusted for stroke and myocardial infarction (MI) fatalities
- The cost of renal monitoring is not being applied to all patients who start treatment with dapagliflozin.
- There are some discrepancies between the submission and the model with regards to the time periods over which some of the costs and utility decrements are applied.
- The process used to sample from beta and gamma distributed parameters within the probabilistic sensitivity analysis (PSA) did not produce appropriately distributed samples

The report also provides further details on some aspects on the model that are not documented in the submission. These were as follows;

- The probability of an event occurring during a cycle is calculated as the difference between the output of the event equation for the current time and the output of the event equation at the previous cycle.
- Treatment discontinuations result in the patient switching treatment immediately and incurring no costs or health benefits from that treatment except for the discontinuation cost.
- The utility gain associated with body mass index (BMI) changes within the PSA model is based on the BMI profile generated using mean parameter values whilst the rest of the simulation uses a BMI profile that is sampled within the PSA.

The DSU were able to compile the DLLs from the source code provided. The diab2sampling DLL compiled by the DSU produced results identical to the diab2sampling DLL provided in the original submission. The DSU were unable to reproduce PSA results which matched exactly those reported in the submission using the diab2sampling DLL, but the ICER generated by the DSU did not vary substantially from that reported in the submission and the differences may have arisen due to differences in the steps taken to set-up the PSA by the DSU as this required multiple changes to be made to the spreadsheet model provided. The diabetes2 DLL compiled by the DSU, which runs the model using mean parameter values, did not produce the results reported in the submission, and did not produce the results generated when using the diabetes2 DLL provided with the original submission. Furthermore, it did not appear to have reached a stable estimate of the incremental QALYs gained after 100 runs. The DSU also noted that the results generated by the diab2sampling DLL when all parameters are set to their mean value did not match those generated by the diabetes2 DLL which uses mean parameter values.

The DSU do not have confidence in the results produced by the model in its current form and would suggest that the following should be addressed as a priority. The version of the model that uses mean parameter values should produce results comparable to that produced by the PSA when all parameters within the PSA are set to their mean value. The process used to sample beta and gamma distributed parameters should be corrected to produce distributions

consistent with those specified in the submission. The data used in the PSA for change in BMI from baseline should be based on the sampled BMI profiles. The DSU would also wish to see further clarification regarding the discrepancies identified between the event and risk factor equations implemented in the model and those reported in the manufacturer's response to the clarification letter.

CONTENTS

1. INTRODUCTION 9
1.1. BACKGROUND 9
1.2. METHODS 9
1.3. STRUCTURE OF THIS REPORT 10
2. MODEL LOGIC 10
2.1. MODEL VERSIONS 11
2.2. PATIENTS 11
2.3. PROGRESSION OF TIME WITHIN THE MODEL 12
2.4. PROCESS OF SIMULATING MULTIPLE PATIENTS ACROSS MULTIPLE RUNS 12
2.5. SIMULATION PROCESS FOR EACH INDIVIDUAL 13
2.5.1. Treatment switches 13
2.5.2. Updates prior to evaluating events 15
2.5.3. Evaluating events for the current cycle 15
2.5.4. All cause mortality 19
2.5.5. Adverse events 19
2.5.6. Hypoglycaemic events 19
2.6. RELATIONSHIP BETWEEN UTILITIES APPLIED AND QALYS GAINED 20
2.7. COSTS APPLIED FOLLOWING DIABETES COMORBIDITIES 20
2.8. PROBABILISTIC SENSITIVITY ANALYSIS 20
3. CONCERNS AND ERRORS IN THE MODEL LOGIC 21
3.1. TIME PERIOD FOR PROBABILITIES GIVEN BY EVENT EQUATIONS 21
3.2. INCONSISTENCIES BETWEEN EVENT EQUATIONS REPORTED AND THOSE APPLIED. 22
3.3. INCONSISTENCIES BETWEEN THE RISK FACTOR EQUATIONS REPORTED AND THOSE EVALUATED IN THE PSA 23
3.3.1. SBP risk factor equation 23
3.3.2. TC:HDL risk factor equation 24
3.3.3. Evolution of weight 24
3.4. DELTA BMI VALUES IN PSA 25
3.5. RENAL MONITORING 25
3.6. UTILITY DECREMENT FOR ADVERSE EVENTS 25
3.7. PATIENT COHORT SIZE 26
4. VALIDATION EXERCISES 27
4.1. COMPARISON OF DIFFERENT DLL VERSIONS SUPPLIED WITHIN THE SUBMISSION 27
4.2. REPRODUCIBILITY OF THE RESULTS USING DLLS GENERATED BY THE SOURCE CODE 28
4.2.1. Diabetes2 DLL: model results using mean parameter values 28
4.2.2. Diab2Sampling DLL: model results with no parameters sampled 30
4.2.3. Diab2Sampling DLL: PSA results 30
4.3. COMPARABILITY OF RESULTS GENERATED BY THE DIABETES2 AND DIAB2SAMPLING DLLS 31
4.4. PARAMETER SAMPLING WITHIN THE PSA VERSION OF THE MODEL 32
4.5. RANDOM NUMBER GENERATION 34
5. CONCLUSIONS 35
6. REFERENCES 37
APPENDIX 1: FLOW CHARTS SHOWING MODEL LOGIC 38
SEE ACCOMPANYING PDF FILE. 38
APPENDIX 2: RANDOM NUMBER GENERATOR TESTS 38
APPENDIX 3: PROBABILISTIC SENSITIVITY ANALYSIS 42

TABLES & FIGURES

Table 1: Event equations for diabetes complications 17
Table 2: Fatality equations 18
Table 3: Comparison of cost-effectiveness results generated using source code provided against those reported in the submission (for add-on to metformin model using mean parameter values with 100 runs of 30,000 patients) 30
Table 4: Comparison of cost-effectiveness results generated using source code provided against those reported in the submission (Dapagliflozin vs SU for add-on to metformin model using PSA with 1,000 runs of 30,000 patients) 31

Table A 1: Diagnostic output from the Dieharder random number testing suite 39

Figure 1: Simulation process for an individual (flow charts for the whole simulation can be found in Appendix 1) 14
Figure 2: Flow chart showing process of evaluating amputation events (similar figures for the other events can be found in appendix 1) 16
Figure 3: Example weight profile for one sampled patient 24
Figure 4: Variation in incremental QALYs over run size for DLLs provided in the original submission (for add-on to metformin model comparing dapagliflozin against SU using mean parameter values) 26
Figure 5: Variation in incremental cost over run size for DLLs provided in the original submission (for add-on to metformin model comparing dapagliflozin against SU using mean parameter values) 27
Figure 6: Variation in incremental QALYs over run size for DLLs generated from source code provided in response to clarification request (for add-on to metformin model comparing dapagliflozin against SU using mean parameter values) 29
Figure 7: Variation in incremental costs over run size for DLLs generated from source code provided in response to clarification request (for add-on to metformin model comparing dapagliflozin against SU using mean parameter values) 29
Figure 8: Comparison of parameter samples against specified distribution for the beta distributed parameter 'MI utility decrement' 33

Figure A 1: Visual representation of the random number generator applied within the simulation (1024 x 1024 pixels) 38
Figure A 2: Comparison of parameter samples against specified distribution for the normally distributed parameter 'HbA1c change' 42
Figure A 3: Comparison of parameter samples against specified distribution for the gamma distributed parameter 'cost of fatal MI' 43
Figure A 4: Comparison of incremental QALYs when using the 'run model with mean values' option and when using the 'run probabilistic sensitivity analysis' option with no parameters sampled 43
Figure A 5: Comparison of incremental costs when using the 'run model with mean values' option and when using the 'run probabilistic sensitivity analysis' option with no parameters sampled 44

ABBREVIATIONS AND DEFINITIONS

AE	Adverse events
BMI	Body mass index
BMS/AZ	Bristol-Myers Squibb / AstraZeneca
CHF	Congestive heart failure
DES	Discrete event simulation
DLLs	Dynamic link library
DPP-4	Dipeptidyl peptidase-4
HbA1c	Glycosylated haemoglobin
HDL	High density lipoprotein
HFS	Hypoglycaemia fear score
ICER	Incremental cost-effectiveness ratio
IHD	Ischaemic heart disease
MS	Manufacturer submission
MI	Myocardial infarction
MRCL	Manufacturer response to the clarification letter
PLS	Patient level simulation
PSA	Probabilistic sensitivity analysis
PVD	Peripheral vascular disease
SBP	Systolic blood pressure
SU	Sulphonylurea
TC	Total cholesterol
TZD	Thiazolidinedione
UKPDS	United Kingdom Prospective Diabetes Study
UTI	Urinary tract infection
VBA	Visual basic application
QALY	Quality adjusted life-years

1. INTRODUCTION

1.1. BACKGROUND

NICE is conducting a single technology appraisal of dapagliflozin (FORXIGA®) for the treatment of type 2 diabetes. The economic evaluation included within the Bristol-Myers Squibb / AstraZeneca (BMS/AZ) submission to NICE is based on a patient level simulation (PLS) which simulates treatment sequences and health outcomes for individual patients and estimates the average costs and benefits over a large cohort of patients. The model is constructed with an Excel spreadsheet front-end which tabulates data which are processed via:

- Excel formulas
- Visual Basic (VBA) functions and macros
- C++ programming code compiled into Dynamic Link Libraries (DLLs)

The VBA within the Excel spreadsheet generates the data arrays which form the inputs for the PLS. The VBA then runs the PLS by calling the DLL which implements the C++ program. Finally the VBA records the output arrays from the PLS within the Excel spreadsheet. The DLL components of the economic model are bespoke pieces of software and are therefore not considered to fall within the list of standard software packages usually accepted by NICE.

The Decision Support Unit was asked by NICE to examine the executable model submitted by Bristol-Myers Squibb / AstraZeneca with the following aims;

1) To establish whether the model function as described by the manufacturer
2) To report any important aspects of the model that are not described in the manufacturer submission
3) To examine whether the C++ code follows the steps described by the manufacturer and uses the data described in the submission
4) To validate whether the model produces the results described in the submission

1.2. METHODS

The DLLs are “black boxes” of code that cannot be investigated through Excel. C++ code has been used to create these DLL’s in a process called “compilation”. By following the process

of compilation again or "recompiling" this C++ code, it is possible to watch how it works in a similar way to validating VBA code in Excel.

The DSU examined the input variables used by the C++ program, the program logic defined within the C++ code and the output variables returned by the program to the spreadsheet model. These were compared against the data and methods reported in the manufacturer submission (MS), and in the manufacturer response to the clarification letter (MRCL), to identify any inconsistencies. The model structure was examined by a health economist in collaboration with a programming consultant to identify any logical inconsistencies within the model structure. A description of the model logic is provided in Appendix 1 to supplement that given in the MS. The source code provided was also compiled into DLLs and checks were made to determine whether these DLLs produced results consistent with those report in the MS.

1.3. STRUCTURE OF THIS REPORT

The second chapter of this report contains a general description of the model which is provided to supplement the details supplied within the manufacturer's submission. The third chapter of this report documents the factual errors which we have found within the simulation model. It also describes any areas of concern identified within the model which could undermine the validity of the model. The fourth chapter describes the validation exercises we have conducted.

2. MODEL LOGIC

The simulation model submitted by the manufacturer is a generic model which has some features which are not relevant to the analysis undertaken to support the BMS/AZ submission to NICE. For example, it includes a type 1 diabetes model and the capability to specify interventions and comparators which are not relevant to the scope of this appraisal. This report will only comment on the issues in the model that are relevant to the economic evaluation of dapagliflozin submitted by BMS/AZ.

2.1. MODEL VERSIONS

The manufacturer submitted multiple versions of the spreadsheet model, one for each treatment comparison addressed within the submission. These were grouped into three folders with each folder addressing a different indication for dapagliflozin (add-on to metformin therapy, add-on to insulin therapy, triple therapy). Within each folder there was a copy of five DLLs, each of which can be called by the spreadsheet models.

- **Diabetes2.dll**: This is called when running the model from the demographics sheet with the 'run model using mean values' option selected
- **Diab2Sampling.dll**: This is called when running the model from the demographics sheet with the 'run probabilistic sensitivity analysis' option selected.
- **Diab2Tornado.dll**: This is called when running the model from the 'Tornado' sheet. This version of the model appears to be designed to conduct univariate sensitivity analyses. However it is based on an older version of the model which has substantive differences from the versions above. Given the differences that exist between this model and those used to produce the basecase results, this model was not considered to provide a valid means of conducting sensitivity analysis and has not been examined in detail by the DSU.
- **Diab2User.dll**: This is called when running the model with the 'run with user data' option selected. This was not considered to be relevant to any of the results presented within the submission and was not examined by the DSU.
- **Diabetes1.dll**: This was identified in the MRCL as being redundant in the context of this submission and was not examined by the DSU.

The description of the model that follows relates to the version of the model which operates when the 'run model using mean values' option is selected, unless otherwise stated.

2.2. PATIENTS

The model simulates the flow of individual patients in a stochastic manner allowing individual patients to have different trajectories through the model giving heterogeneity in the events experienced, and the costs and benefits accrued by individual patients. In the manufacturer's basecase analysis, all the simulated patients have identical characteristics at baseline giving a uniform cohort of patients who vary only in the events they experience once they enter the simulation. Even the gender, ethnicity and smoking status for a simulated individual is defined as a proportion, reflecting the average across the cohort, rather than as a

binary variable reflecting the status of the individual. For example, gender is defined on a scale of 0 to 1 for each individual rather than being defined as either male (0) or female (1).

The only patient characteristic that can vary between patients at baseline is the patient's history of diabetes complications, but in the basecase analysis it is assumed that no patient has a history of previous diabetes complications. Scenario 13 described on page 245 of the MS allows individuals to have a prior history of diabetes complications and these are sampled on an individual basis allowing individuals within the cohort to have different characteristics at baseline.

2.3. PROGRESSION OF TIME WITHIN THE MODEL

The economic model is described within the submission as a discrete event simulation (DES). Normally within a DES, time is treated as a continuous variable and time is only progressed within the simulation when an event occurs. In a DES a list of possible events that can happen is maintained, with the time to each event being sampled and the simulation progressing to the time of the next event due to occur. Whilst this model is described as a DES (page 201 of the MS), it is actually simulating the events which occur within fixed cycle lengths of 6 months. So whilst the model simulates the trajectory of an individual patient, it differs from what would normally be understood as a DES.

2.4. PROCESS OF SIMULATING MULTIPLE PATIENTS ACROSS MULTIPLE RUNS

The model simulates the progress of one individual patient across either the whole model timeframe or the whole lifetime of the patient, whichever is shorter, and then goes on to simulate the next patient. Once it has simulated all the individuals within the cohort (30,000 in the manufacturer's basecase analysis) it then goes on to repeat the process for the next model 'run'. There are 100 runs in the manufacturer's basecase analysis which uses the mean parameter values. In this analysis neither the baseline characteristics nor the parameter values vary from one run to the next. The only variation between these model runs comes from the random number sequence which is used to sample the event probabilities which are applied each cycle. These 100 runs of 30,000 patients effectively produce a patient cohort of 3 million for the manufacturer's baescase analysis using mean parameter values.

2.5. SIMULATION PROCESS FOR EACH INDIVIDUAL

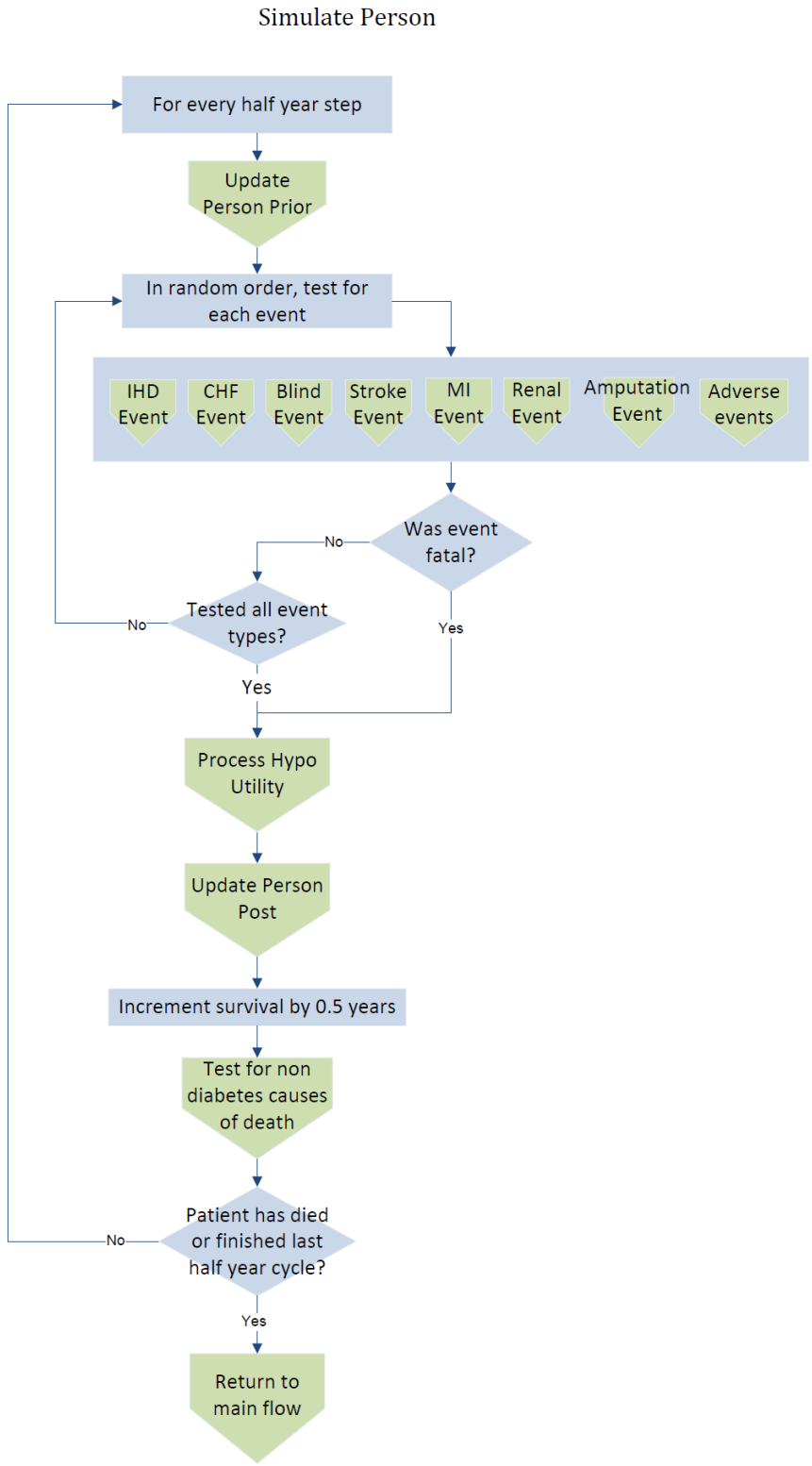
The simulation process for an individual is shown in Figure 1. (This is taken from a series of linked flow charts describing the whole simulation process which can be found in Appendix 1.)

2.5.1. Treatment switches

In the very first cycle, the process denoted by 'update person prior' in Figure 1, checks to see if the individual discontinues the first treatment and if so it switches the person to the next treatment immediately. This check is only made within 'update person prior' in the first cycle and thereafter the process 'update person post' is used to check for treatment switches. There are two possible ways to switch treatment. Once a year, the process 'update person post', checks to see if the person has reached the glycosylated haemoglobin (HbA1c) threshold for their current treatment. When a treatment switch occurs as a result of the patient reaching the HbA1c threshold for that treatment, the process 'update person post' immediately checks to see whether the new treatment is discontinued, in which case a second treatment switch occurs. Therefore the second line treatment can be skipped entirely. If the person does not discontinue in their first cycle of a new treatment, they stay on that treatment until they reach the HbA1c threshold.

In the add-on to metformin therapy comparison, non-zero discontinuation rates are specified in Table 58 of the MS for the first line treatments allowing the first line treatment to be skipped, but the discontinuation rate for second line treatments is zero so no patient can skip second line therapy. Discontinuation rates were set to zero for all drugs in the add-on to insulin therapy comparison. In the triple therapy comparison, both the first line treatment (dual therapy), and the second line treatment (triple therapy) have none-zero discontinuation rates allowing a proportion to skip these treatments incurring only the cost of discontinuation and no other costs or health effects from these treatments.

Figure 1: Simulation process for an individual (flow charts for the whole simulation can be found in Appendix 1)



2.5.2. Updates prior to evaluating events

The process ‘update person prior’ is used to apply treatment costs and maintenance costs for events experienced in previous cycles. It also updates the record of utility for that cycle with the ‘current utility’ value from the end of the previous cycle which incorporates any utility decrements that have resulted from events experienced in the previous cycle. It then resets any ‘one-off’ utility decrements that apply to a single cycle.

The process ‘update person prior’ also updates the individual’s risk factors using data from the ‘biannual risk factor inputs’ sheet of the spreadsheet according to the current treatment and the number of cycles experienced on that current treatment. The risk factors updated are;

- Blood glucose (HbA1c)
- Ratio of total to high density cholesterol (TC:HDL)
- Systolic blood pressure (SBP)
- Weight (from which body mass index (BMI) and change in BMI from baseline are calculated)

2.5.3. Evaluating events for the current cycle

After the process ‘update person prior’ has been completed, the model goes on to consider the events that occur in the coming cycle. These can include seven diabetes complications and up to five adverse events (AEs) giving a total of 12 event equations. Random sampling is used to shuffle the order of these events with each event being tested in sequence until either a fatal event occurs or until all 12 have been tested. This process allows multiple events of different types to occur within one cycle. Events of the same type cannot occur twice within the simulation as these are suppressed by the options selected in the basecase analysis.

Figure 2 shows the process used to evaluate individual diabetes comorbidities, in this case amputation. The equation is evaluated for the current time and the difference is then taken between the value of the equation for this cycle and the value stored from the previous cycle. This is then compared to a randomly sampled number, on a scale of 0 to 1, to determine if the event occurs. In the case of ischaemic heart disease (IHD) and congestive heart failure (CHF) events, an additional check is made which prevents these events occurring if a previous myocardial infarction (MI) has been experienced. The event equations are summarized in Table 1.

Figure 2: Flow chart showing process of evaluating amputation events (similar figures for the other events can be found in appendix 1)

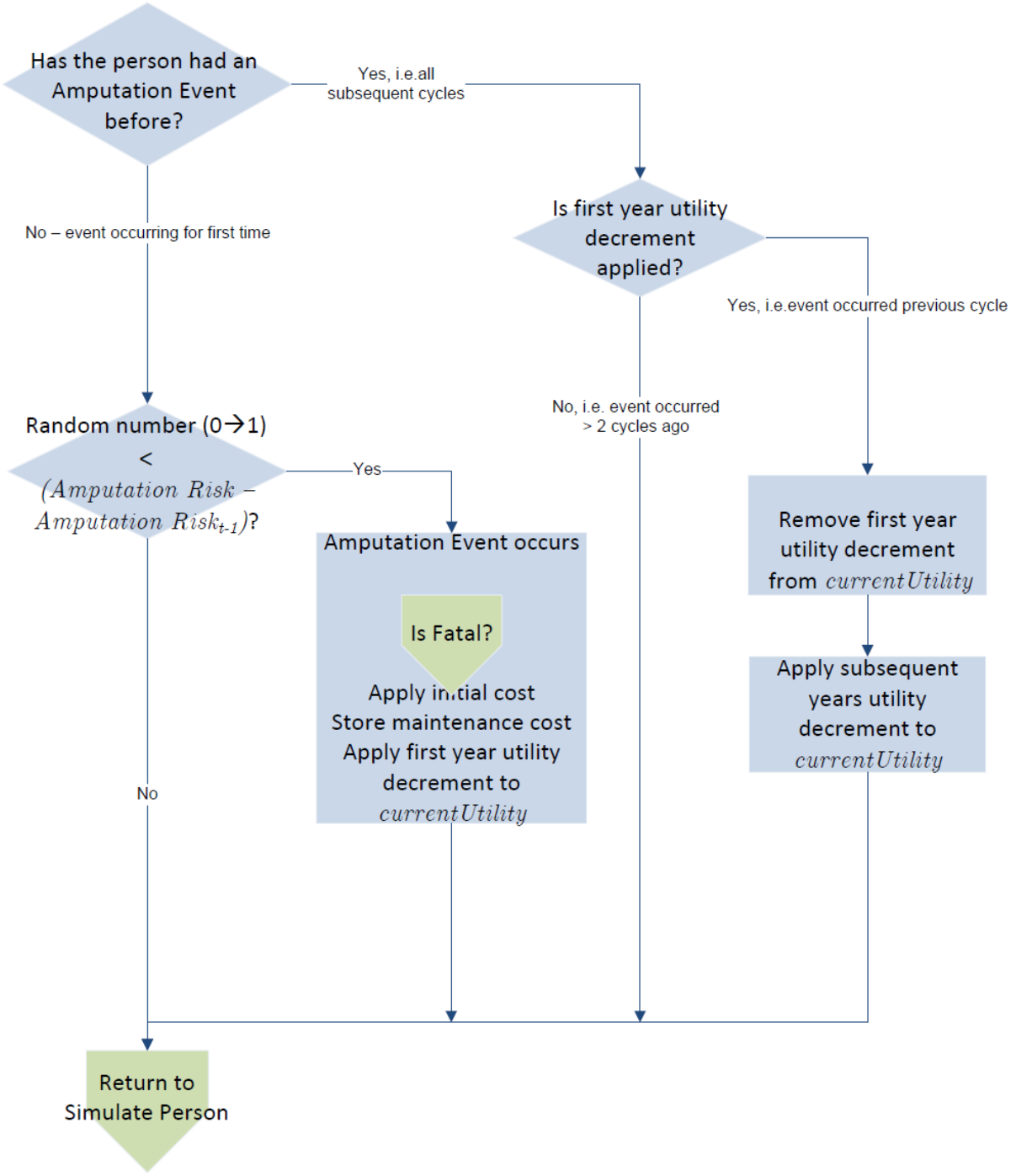


Table 1: Event equations for diabetes complications

Event	Equations used to estimate risk of event
IHD	β = -5.31 + 0.031 * (age at diagnosis - 52.59) - 0.471 * sex + 0.125 * ($HbA1c_t$ - 7.09) + 0.098 * ((SBP_t - 135.09) / 10) + 1.498 * (ln ($tcHDL_t$) - ln (5.23)) *IHD Risk* = e^{β} * $year^{1.150}$ * BMI Risk Factor * CV Risk Factor
MI	β = -4.977 + 0.055 * (age at diagnosis - 52.59) - 0.826 * sex - 1.312 * ethnicity + 0.346 * smoking + 0.118 * ($HbA1c_t$ - 7.09) + 0.101 * ((SBP_t - 135.09) / 10) + 1.190 * (ln ($tcHDL_t$) - ln (5.23)) + 0.914 * IHD Event Occurred + 1.558 * CHF Event Occurred *MI Risk* = e^{β} * $year^{1.257}$ * BMI Risk Factor * CV Risk Factor
CHF	β = -8.018 + 0.093 * (age at diagnosis - 52.59) + 0.066 * (BMI_t - 27.77) + 0.157 * ($HbA1c_t$ - 7.09) + 0.114 * ((SBP_t - 135.09) / 10) *CHFRisk* = e^{β} * $year^{1.711}$ * BMI Risk Factor * CV Risk Factor If Therapy == Rosiglitazone, *CHFRisk* = *CHFRisk* * *CHFRisk Factor*
Stroke	β = -7.163 + 0.085 * (age at diagnosis - 52.59) - 0.516 * sex + 0.355 * smoking + 0.128 * ($HbA1c_t$ - 7.09) + 0.276 * ((SBP_t - 135.09) / 10) + 0.113 * (ln ($tcHDL_t$) - ln (5.23)) + 1.428 * AF + 1.742 * CHF Event Occurred Stroke *Risk* = e^{β} * $year^{1.497}$ * BMI Risk Factor * CV Risk Factor
Amputation	β = -8.718 + 0.435 * ($HbA1c_t$ - 7.09) + 0.228 * ((SBP_t - 135.09) / 10) + 2.436 * PVD + 1.812 * Blind Amputation *Risk* = e^{β} * $year^{1.451}$ * BMI Risk Factor
Renal failure	β = -10.016 + 0.404 * ((SBP_t - 135.09) / 10) + 2.082 * Blind Renal *Risk* = e^{β} * $year^{1.865}$ * BMI Risk Factor
Blindness	β = -6.464 + 0.069 * (age at diagnosis - 52.59) + 0.221 * ($HbA1c_t$ - 7.09) Blind *Risk* = e^{β} * $year^{1.154}$ * BMI Risk Factor

CV=cardiovascular, PVD=peripheral vascular disease, AF=atrial fibrillation

When an event occurs for the first time, the model checks to see if the event is fatal and updates the person's utility value to reflect the 'first year' utility decrement. It also applies the initial cost at this stage. In the first cycle after the event occurs, this 'first year' utility decrement is removed and the 'subsequent year' utility decrement is applied. It should be noted that the switch from 'first year' to 'subsequent year' utility decrement occurs after 6 months and not after 1 year, but in the manufacturer's basecase analysis the 'first year' and 'subsequent year' utility decrements are identical, as specified on the 'utilities' sheet of the

spreadsheet model, so no error is introduced. In all subsequent cycles after an event, the utility value is not updated and so the decrement applied in the second cycle is maintained.
When a CHF, amputation or renal event occurs the model checks the general event fatality equation to determine whether the event is fatal. When a stroke or MI event occurs the model checks fatality equations that are specific to those events. There is no check for fatality following blindness or IHD events.

The fatality equations are summarized in Table 2. The appropriate fatality equation is checked for each event that occurs before progressing to the next event in the event list. Therefore it is possible for the general event fatality equation to be evaluated multiple times within one cycle. Once a fatal event occurs, no further event equations are evaluated but the hypoglycaemic events for that cycle are still processed.

The initial cost for any events occurring in the same cycle as a fatal event are accrued, but the utilities are not updated as the fatal event prevents the patient returning to the process 'update person prior' which implements utility decrements from events that occurred in the previous cycle.

Table 2: Fatality equations

Event	Equations used to estimate risk of event being fatal
MI	$\beta = 0.713 - 0.048 * (age\ at\ diagnosis - 55) - 0.178 * (HbA1c_t - 7.09 + 0.23) - 0.141 * (SBP_t - 141)/10 - 0.104$ $MI\ Fatal\ Risk = 1/(1-\exp(\beta))$
Stroke	$\beta = 1.684 - 0.249 * (SBP_t - 141)/10$ $Stroke\ Fatal\ Risk = 1/(1-\exp(\beta))$
CHF/ renal failure / amputation	*If* $(age\ first\ event - 52.59) > 0$ $logAgeEvent = \ln(age\ first\ event - 52.59)$ *Else* $logAgeEvent = 0$ $\beta = -3.251 + logAgeEvent + 0.114 * (HbA1c_t - 7.09) + 2.64 * MI + 1.048 * Stroke$ $Fatal\ Risk = \exp(\beta)/(1+\exp(\beta))$
Non diabetes related death	$\beta = (1 - sex) * LifeTables[Male][age] + LifeTables[female][age]$ $ProbabilityOfDeath = 1 - (1-\beta)^{0.5} - DiabetesProbabilityDeath$ Note: *DiabetesProbabilityDeath* is the accumulated probability of death from CHF, Amputation and Renal Failure in this cycle. The value is reset in *UpdatePersonPrior*.

2.5.4. All cause mortality

In addition to fatality occurring as a result of diabetes related events, the model also includes all cause mortality events which are estimated separately from life-tables. Male and female specific data is applied using a weighted average according to the proportion of females within the cohort at baseline. This approach assumes that the ratio of males to females within the cohort is constant over time and would therefore not be appropriate if the overall survival of males and females within the model is different as this would result in a ratio which varies over time.

The risk of death taken from the life-tables is adjusted to account for the risk of diabetes related death from CHF, renal events and amputations in that cycle, such that the overall risk of death from CHF, renal events, amputations and non-diabetes related death is equal to that given in the life-table. No adjustment is made to account for the risk of fatality from stroke or MI events. This differs from the description of all-cause mortality on page 213 of the MS where it states that both cardiovascular and diabetes related deaths were subtracted from all-cause mortality.

2.5.5. Adverse events

In the manufacturer's basecase analysis, the first adverse event (AE) category is used to apply the cost of renal monitoring when dapagliflozin is initiated. The second and third AE categories are used to model urinary tract infections (UTI) and genital infections. The fourth and fifth AE categories are not used in the submitted analysis. Hypoglycaemic events are modeled using a separate process which occurs after the diabetes comorbidity and other AEs have been tested.

2.5.6. Hypoglycaemic events

The equations determining the rate of severe and non-severe hypoglycaemic events are evaluated every 6 month cycle and the probability of experiencing a symptomatic or severe hypoglycaemic event is half the value specified in Table 58 of the MS. It therefore appears that the values in Table 58 are annual probabilities that are adjusted to give the probability per cycle.

In the basecase analysis, the utility decrement associated with hypoglycaemic events is estimated based on a fear of hypoglycaemia score (HFS). The equation applied to estimate

the HFS and the resultant utility decrement is as reported under question B28 of the MRCL. However, the utility decrement associated with fear of hypoglycaemia is not halved in the same manner as the utility decrements associated with diabetes comorbidites.

2.6. RELATIONSHIP BETWEEN UTILITIES APPLIED AND QALYS GAINED

The model tracks the current utility value for an individual patient which is updated when various events, such as diabetes comorbidity events, occur within the model. The current utility value for each model cycle is recorded within the process 'update person prior' giving an array of utility values for each individual across all the time cycles. These are then summed within the process 'compute sum utilities'. The baseline utility, which applies to patients who have not experienced any events, and the utility decrements applied following each of the diabetes comorbidities are halved compared to those reported in Table 60 of the MS. Our interpretation of these steps is that halving the utilities and then summing the utility values across the model cycles is equivalent to calculating the QALYs gained as the cycle length is half a year.

2.7. COSTS APPLIED FOLLOWING DIABETES COMORBIDITIES

The costs in Table 63 of the MS are described as being 'annual direct medical complication costs'. However, it should be noted that the costs labeled 'fatal' and 'non-fatal' are applied in full immediately following an event, whilst the costs labeled as 'maintenance' which are applied in subsequent cycles are halved before being applied. This suggests that the initial costs are considered to be one-off costs incurred in full, whilst the maintenance costs are considered to be annual costs and are therefore adjusted to reflect the 6 month cycle length. It should be noted that the maintenance costs are applied from 6 months after the initial event and so the patient accrues the initial cost and 6 months of maintenance costs in the first year after an event.

2.8. PROBABILISTIC SENSITIVITY ANALYSIS

The VBA and C++ code implemented when running the model with the 'run probabilistic sensitivity analysis' option selected is completely separate from that implemented when running the model with the 'run model using mean values' option selected. The code was examined to identify the main differences in processes and structure between the two models which are described here. However, it wasn't possible for the two versions to be exhaustively

compared in the time frame available. Validation exercises were performed to determine whether the two versions of the model are equivalent and these are described in section 4.

In the probabilistic sensitivity analysis (PSA), which uses 1,000 cohorts (runs) of 30,000 patients, the parameter values are varied from one run to the next. The structure of the probabilistic sensitivity analysis allows the baseline characteristics to be sampled from one run to the next but baseline characteristics are not sampled in the PSA reported in the MS (see Table 72 of MS and question B13 of the MRCL).

The sampling of parameter values occurs within the DLL based on the mean, standard deviation and maximum and minimum values specified on the 'PSA map' sheet of the spreadsheet model. This sheet is also used to specify which parameters are to be sampled within the PSA.

In the version of the model that runs the probabilistic sensitivity analysis, the risk factor evolution (HbA1c, SBP, TC:HDL and BMI) is evaluated using equations within the simulation itself rather than within the spreadsheet model as these are dependent on parameters which are sampled at the start of the simulation. It also estimates treatment costs for insulin as these are dependent on BMI.

No other differences in model logic were identified between the PSA version of the model and the version that runs with mean parameter values from examining the source code provided.

3. CONCERNS AND ERRORS IN THE MODEL LOGIC

3.1. TIME PERIOD FOR PROBABILITIES GIVEN BY EVENT EQUATIONS

The diabetes comorbidity and AE event equations are evaluated once every 6 month cycle. Initially the DSU could not identify how the model adjusts the comorbidity event equations to take into account the cycle length. The DSU asked for clarification from the manufacturer on this matter. The manufacturer response stated, "The time period for which the probabilities returned from the event equations shown in Table 4 [of the MRCL] are dependent on the time

parameter passed to the equation. As the model works in 6-monthly cycles, time is passed to these equations in half-yearly increments, for example the difference between cumulative event rate at year 1 versus year 1.5 will give the 6-monthly probability required". Following this clarification, the DSU were able to confirm that the model does take the difference between the value of the equation for the current cycle and the value from the previous cycle as described in section 2.5.3 above. It would therefore appear that the equations are being interpreted as giving the cumulative hazard, and the difference between the cumulative hazards from one cycle to the next is being taken to give the probability of the event occurring during the cycle. The DSU noted that the probability of a transition occurring during cycle length u is given by;

Transition probability = 1-exp { H(t-u) – H(t) }

where H(t) is the cumulative hazard at time t. This would only be equivalent to the approach taken by the manufacturer if the event equations, as stated in Table 1 above, do in fact give the cumulative hazard observed in UKPDS 68[1] and if the term, *{ H(t-u) – H(t) },* is sufficiently small for the approximation, *1-exp(x) ~ x,* to be made.

3.2. INCONSISTENCIES BETWEEN EVENT EQUATIONS REPORTED AND THOSE APPLIED.

The DSU compared the event equations in Table 4 of the MRCL with those specified in the C++ code. The following discrepancies were noted:

- In the IHD, MI and stroke equations, the term

  log(total:hdl-5.23)

  is being evaluated as

  log(total:hdl) – log(5.23).
- In the fatality event equation, the term

  If ((AGE_EVENT – 52.59) < 0)

  Is set to zero when *AGE_EVENT is <52.59*

The DSU did not attempt to amend these equations to match those specified in Table 4 as this would result in the model attempting to evaluate the logarithm of a negative number, which would result in a model error.

The fatality equation applied for CHF, amputation and renal events matches that reported in Table 4 of the MRCL (question B10), but the equation in Table 4 appears to differ from that reported in United Kingdom Prospective Diabetes Study 68 (UKPDS 68[1]) which is cited as the source of this equation. Also in the CHF equation, the current BMI value is being used to evaluate the risk of CHF rather than BMI at diagnosis as specified in UKPDS 68.[1] Table 4 of the MRCL doesn't specify whether this is BMI at time of diagnosis or current BMI.

3.3. INCONSISTENCIES BETWEEN THE RISK FACTOR EQUATIONS REPORTED AND THOSE EVALUATED IN THE PSA

The DSU compared the event equations in Table 3 in the MRCL with those specified in the C++ code for the PSA version of the model. The following discrepancies were noted.

3.3.1. SBP risk factor equation

The equation for year 2 SBP is being evaluated in the code as;

IF (months benefit = 12)

THEN sbp(2) = 0.03+0.039xlog(2+duration of diabetes)

+0.717(sbp(1)-135.09)/10+0.127x(**BASE**-135.09)/10)x10+135.09*

ELSE sbp(2) = baseline + (level change for defined therapy)

but is given in Table 3 of the MRCL as;

IF (months benefit = 12)

THEN sbp(2) = 0.03+0.039xlog(2+duration of diabetes)

+0.717(sbp(1)-135.09)/10+0.127x(**sbp(1)**-135.09)/10)x10+135.09*

ELSE sbp(2) = baseline + (level change for defined therapy)

These would be equivalent if BASE=SBP(1), as stated in Table 3 of the MRCL, which is true for the first therapy in the treatment sequence. However, the description of the SBP risk factor derivation in Table 3 of the MRCL also states that the baseline SBP value for therapies 2 and 3 should be updated to reflect the value of SBP at the year of therapy switch. However, the code applies the baseline SBP value from the start of the model for the parameter named 'BASE' in Table 3 of the MRCL and not the baseline SBP value at the start of each therapy (which is named 'SBP(1)' in Table 3 of the MRCL). Therefore, the equation evaluated in the code doesn't match the equation in Table 3 of the MRCL once the patient switches to the second or third therapy. Given that SBP rises over time and the rate of increase is positively correlated with the baseline value, this error is likely to result in later SBP values being underestimated.

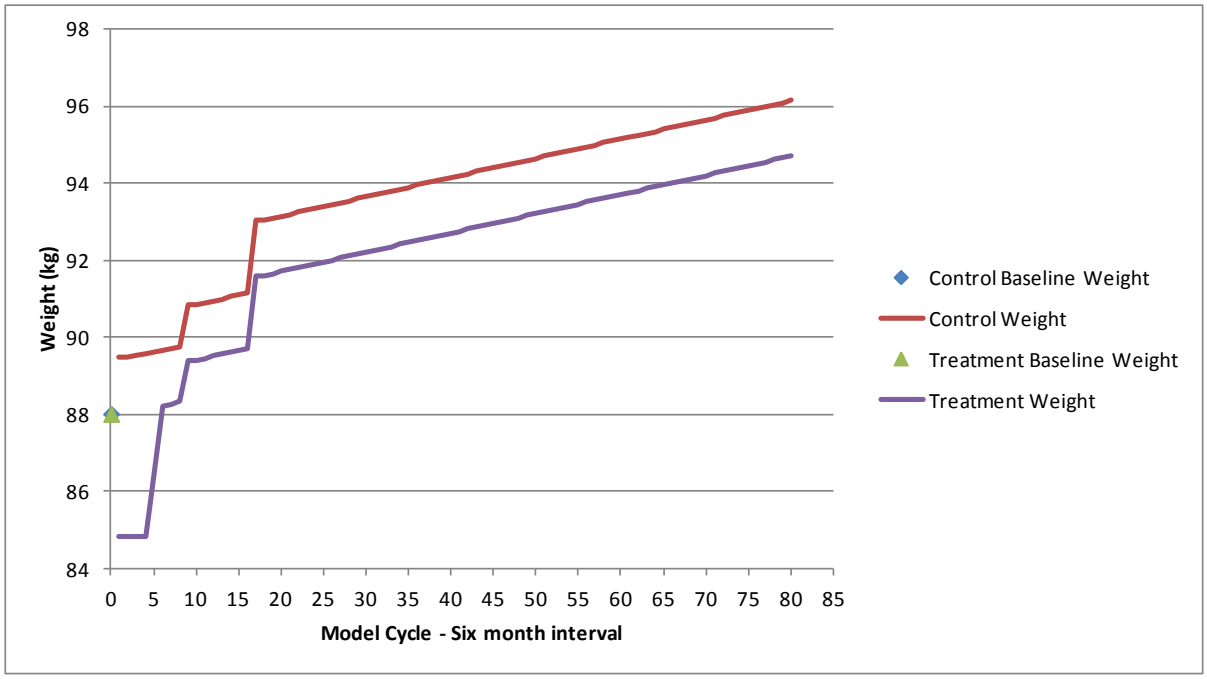
3.3.2. TC:HDL risk factor equation

The TC:HDL equation is correct provided that the parameter 'months benefit' is equal to 12 which it is in the submitted analyses. At 6 months the TCL:HDL equation value is set equal to the value for 1 year. From that point onwards the value for cycles that fall between whole years is interpolated from the annual values.

3.3.3. Evolution of weight

The evolution of weight within the model is more complicated than that specified by the equations provided in Table 3 of the MRCL because it allows for an initial treatment effect, a period of maintained benefits and a catch-up period for the first treatment. The weight evolution applied within the PSA model is consistent with the description given on page 212 of the MS except that the simulation applies the full treatment effect immediately and not gradually over the course of the first year as implied by Figure 27 of the MS. The treatment modified weight achieved on dapagliflozin is maintained for four cycles (2 years), provided a treatment switch does not occur. An illustration of the values applied for one example patient is given in Figure 3 below. This is consistent with the data used when running the model with mean values, as the array of BMI values taken from the 'Biannual risk factor input sheet' also starts with four identical values for patients on dapagliflozin.

Figure 3: Example weight profile for one sampled patient



3.4. DELTA BMI VALUES IN PSA

In the PSA version of the model, the BMI profiles are evaluated within the simulation after the treatment related weight gain has been sampled from its distribution. There is a utility applied within the model which is dependent on 'delta BMI' the difference between the patient's current BMI and their starting BMI. However, in the PSA version of the model, these 'delta BMI' values are being pulled into the model from the values in the 'Biannual risk factor input sheet' which are based on the BMI profiles generated when using the mean parameter value for the treatment related weight gain. Therefore, they do not reflect the BMI profile which has been sampled within the PSA. This is a logical inconsistency, which will result in the uncertainty around the quality adjusted life year (QALY) gain being underestimated, but there may be no bias in the mean QALY gain as the treatment related weight gain is normally distributed.

3.5. RENAL MONITORING

The model does not apply the cost of renal monitoring when dapagliflozin treatment is initiated to all patients in the intervention arm. The probability of experiencing the 'adverse event' of renal monitoring is set to 0.5 but is evaluated twice with a one-off cost of £38.67 and no utility decrement. Due to the ordering of the events within the model, this event is not evaluated for every patient. In a cohort of 10,000 patients we found that 203 patients discontinued dapagliflozin and moved onto the next treatment before this event was evaluated and 24 patients died without having experienced this event. The average cost of renal monitoring in this cohort of 10,000 patients was £37.43 per patient. Whilst this does appear to be a logical error within the model, the DSU considered it unlikely that it would have a significant impact on the incremental cost-effectiveness ratios (ICERs).

3.6. UTILITY DECREMENT FOR ADVERSE EVENTS

The utility decrement applied in the model for the adverse events of UTI and genital infection are applied as a one-off decrement in the cycle that the event happens. However, whilst it is only applied for one cycle, the full decrement given in Table 60 of the MS is applied.

All other utility decrements, such as those that apply after a comorbidity event, are halved before being applied within the model and the baseline utility value is also halved. Our interpretation of this logic is that this is done to convert the utility values for each cycle to QALYs accrued as the time cycle is half a year. It therefore appears that these one-off utility

decrements for adverse events are being applied as a one-off QALY decrement which is accrued within 6 months of the event being experienced rather than as a utility decrement which is applied for either 6 months or 1 year.

3.7. PATIENT COHORT SIZE

The model is effectively estimating average outcomes across 100 runs of 30,000 patients within the basecase analysis which uses mean parameter values. Figures 4 and 5 show the mean incremental costs and QALYs when increasing the patient cohort from 30,000 (1 run) to 6 million (200 runs) for the comparison of dapagliflozin against sulphonylurea (SU) for the add-on to metformin indication. It can be seen that there is the potential for significant bias when the average is estimated over one run of 30,000 patients particularly with regards to the incremental QALYs which do not appear to stabilise until around 150 runs. The number of patients used in the inner loop of the PSA is 30,000 suggesting that the PSA could be biased.

Figure 4: Variation in incremental QALYs over run size for DLLs provided in the original submission (for add-on to metformin model comparing dapagliflozin against SU using mean parameter values)

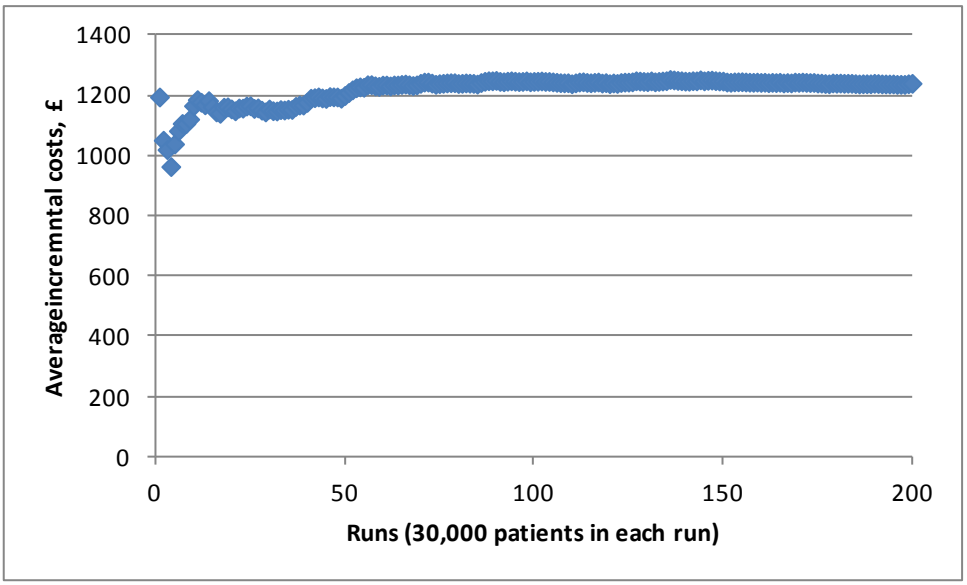
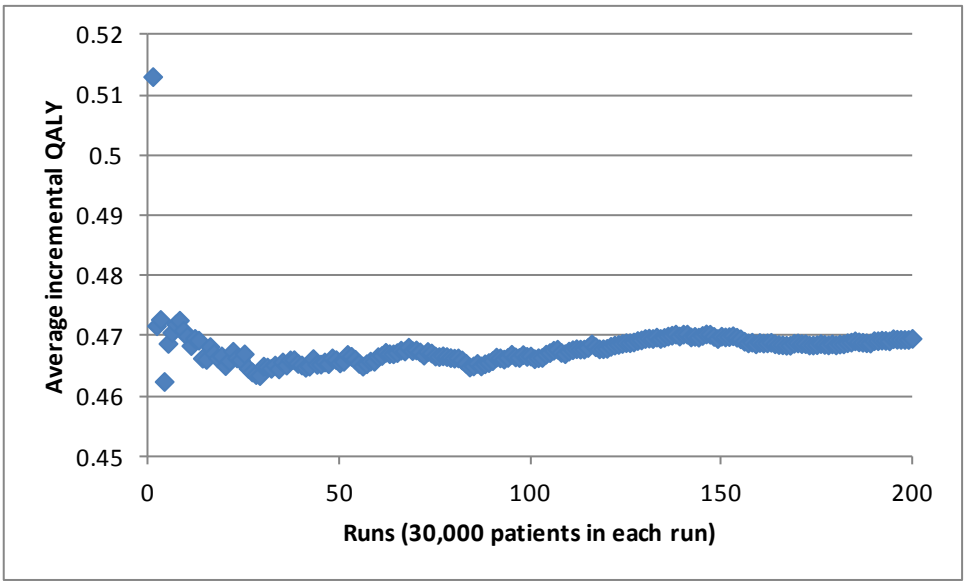


Figure 5: Variation in incremental cost over run size for DLLs provided in the original submission (for add-on to metformin model comparing dapagliflozin against SU using mean parameter values)



4. VALIDATION EXERCISES

4.1. COMPARISON OF DIFFERENT DLL VERSIONS SUPPLIED WITHIN THE SUBMISSION

In the manufacturer submission models were provided in three folders covering the three indications for dapagliflozin (add-on to metformin therapy, add-on to insulin therapy and triple therapy). Each folder contained a separate copy of the DLLs required to run the basecase model using mean values (Diabetes2.dll), and the PSA (Diab2Sampling.dll). The DSU examined these DLLs to assess whether the multiple versions of the DLLs within the separate folders are identical as stated in the MRCL.

The DLLs with the same name were compared against each other using a standard binary file comparison tool, Windiff (windiff.exe is included within the standard support tools provided within Microsoft operating systems). This software can tell you if two files are identical, but cannot indicate the cause of differences in binary files. There were no differences found for the Diabetes2 DLL which is used to run the model with mean parameter values. It was noted that the folder for add-on to metformin therapy comparison had an older version of the Diab2Sampling DLL than the folders for the add-on to insulin therapy comparison and the

triple therapy comparisons. However, running the add-on to metformin model for the dapagliflozin vs SU comparison using the Diab2Sampling DLL from the add-on to insulin therapy folder gave identical results when considering 100 runs of 30,000 patients. Therefore the differences in the Diab2Sampling DLLs provided within the different folders did not appear to have any impact on the results generated.

4.2. REPRODUCIBILITY OF THE RESULTS USING DLLS GENERATED BY THE SOURCE CODE

In the original clarification letter, the manufacturer was asked to provide source code allowing each of the compiled DLLs to be recreated. The DSU re-ran some selected analyses to verify that the source code provided could be compiled into DLLs and that these DLLs could be used by the submitted models to recreate the results reported in the submission.

4.2.1. Diabetes2 DLL: model results using mean parameter values

The Diabetes2 DLL produced by re-compiling the source code provided by the manufacturer did not produce results identical to those produced by the Diabetes2 DLL provided with the original submission. The basecase results when using the Diabetes2 DLL generated from compiling the source code are provided in Figures 6 and 7 for 100 runs of 30,000 patients. It can be seen from Figure 6 that the mean incremental QALY gain has yet to stabilise at 100 runs and appears to be drifting downwards as the run size increases. The incremental costs do appear to have stabilised by 100 runs. Table 3 compares the results generated by these DLLs with those reported in the submission for 100 runs of 30,000 patients. From this it can be seen that the mean incremental costs and QALYs differ, although the ICER is not substantially different. We could not identify why the results generated with the DLL compiled from the source code provided differed from those generated by the DLLs provided with the original submission. We also could not identify why the QALY values did not stabilise within 100 runs.

Figure 6: Variation in incremental QALYs over run size for DLLs generated from source code provided in response to clarification request (for add-on to metformin model comparing dapagliflozin against SU using mean parameter values)

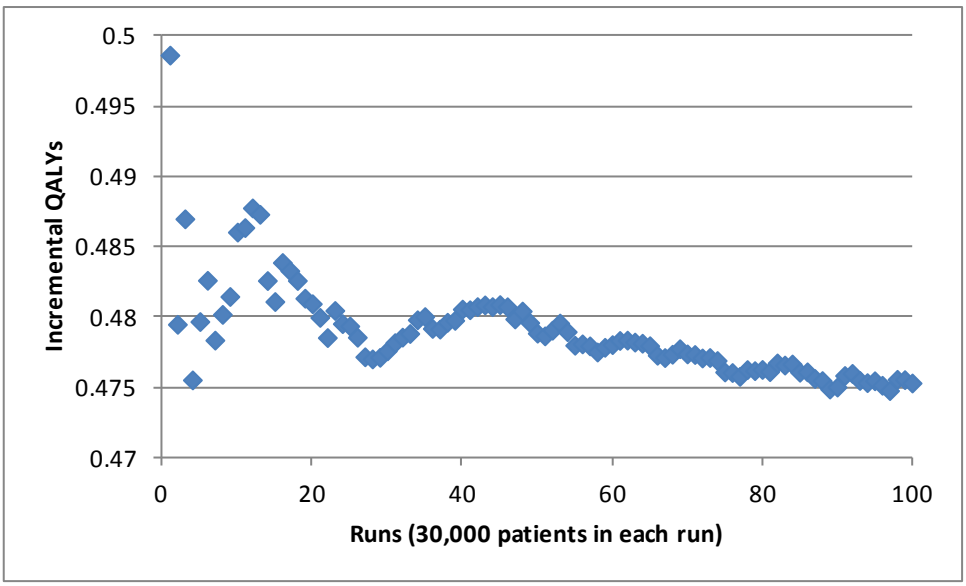


Figure 7: Variation in incremental costs over run size for DLLs generated from source code provided in response to clarification request (for add-on to metformin model comparing dapagliflozin against SU using mean parameter values)

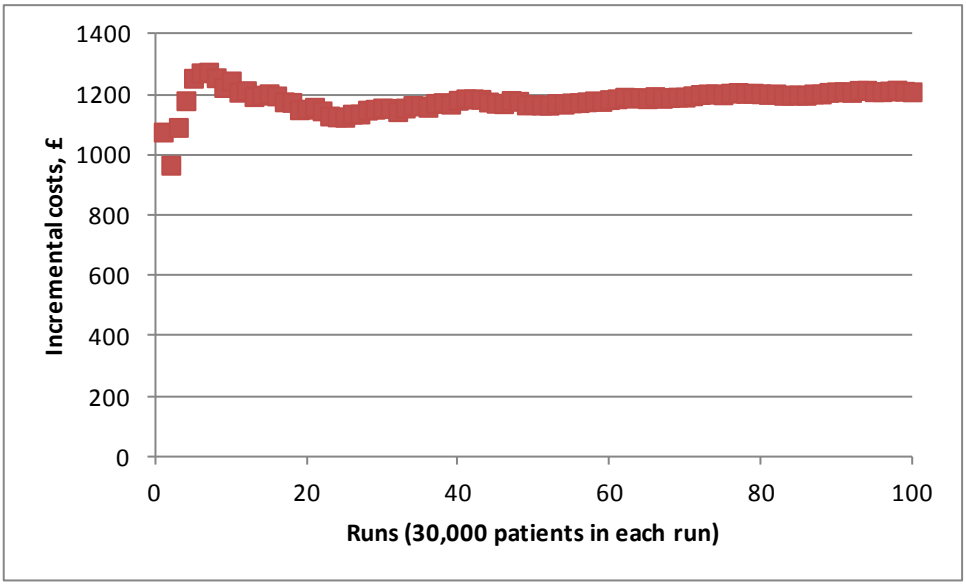


Table 3: Comparison of cost-effectiveness results generated using source code provided against those reported in the submission (for add-on to metformin model using mean parameter values with 100 runs of 30,000 patients)

Comparison	Total			Incremental			ICER (£)
	Costs (£)	LYG	QALYs	Costs (£)	LYG	QALYs	Incremental cost per QALY gained
Results reported in submission							
SU	£ 11,658	14.71	11.28	-	-	-	-
Dapagliflozin	£ 12,904	14.76	11.74	+£ 1,246	+0.050	+0.467	£ 2,671
Results generated using DLLs compiled from source code provided							
SU	£ 11,721	14.67	11.25				
Dapagliflozin	£ 12,930	14.73	11.72	+£1,209	+0.061	+0.475	£2,544

4.2.2. Diab2Sampling DLL: model results with no parameters sampled

The results generated using the DLLs provided with the original submission were compared with those generated using the DLLs compiled from the source code provided. This was done by running the dapagliflozin versus SU model for the add-on to metformin therapy comparison over 100 runs of 30,000 patients. No differences were identified.

4.2.3. Diab2Sampling DLL: PSA results

The DSU attempted to run the model (for dapagliflozin vs SU for add-on to metformin) with those parameters specified in Table 72 of the MS selected to be varied in the PSA. Initially the DSU were unable to run the PSA to completion as the maximum values for the 'HDL-C change' and 'Total-C change' parameters in the comparator arm were lower than the minimum values which caused the program to enter an infinite loop. However, once these values were reversed the PSA ran to completion. The results generated by the DLL compiled from the source code provided were identical to those produced by the original DLL supplied with the submission (based on 1000 runs of 30,000 patients). However, these results were not identical to those reported in Table 90 of the MS, as shown in Table 4 below. The ICERs using the mean costs and QALYs are £2,668 per QALY for the estimates reported in the submission and £2,242 per QALY for the estimates obtained by the DSU running the PSA.

Table 4: Comparison of cost-effectiveness results generated using source code provided against those reported in the submission (Dapagliflozin vs SU for add-on to metformin model using PSA with 1,000 runs of 30,000 patients)

	Results reported in Table 90 of MS			Results obtained by DSU		
Outcome	**Point estimate**	**LL 95%CI**	**UL 95%CI**	**Point estimate**	**LL 95%CI**	**UL 95%CI**
ΔQALYs	0.467	0.420	0.665	0.512	0.363	0.661
ΔCosts	£ 1,246	£ 613	£ 1,637	£1,148	£1,105	£1,192

These differences may be due to the fact that it was necessary for the DSU to make 70 changes to the PSA sheet in order to set up the PSA analysis. Forty-seven of these were indicators selecting the required variables. There were 11 parameters which required standard deviations to be inputted (as directed in the User guidance DCEM document provided in the MS) and two which required their maximum and minimum parameter values to be reversed for the PSA to run to completion (as described above). The DSU also amended the standard errors for the 8 utility values to those quoted in Table 72 of the MS as the values in the spreadsheet did not match those quoted. If any one of these changes differed from the steps taken by the manufacturer when setting up the PSA, the results could vary to the degree shown in Table 4.

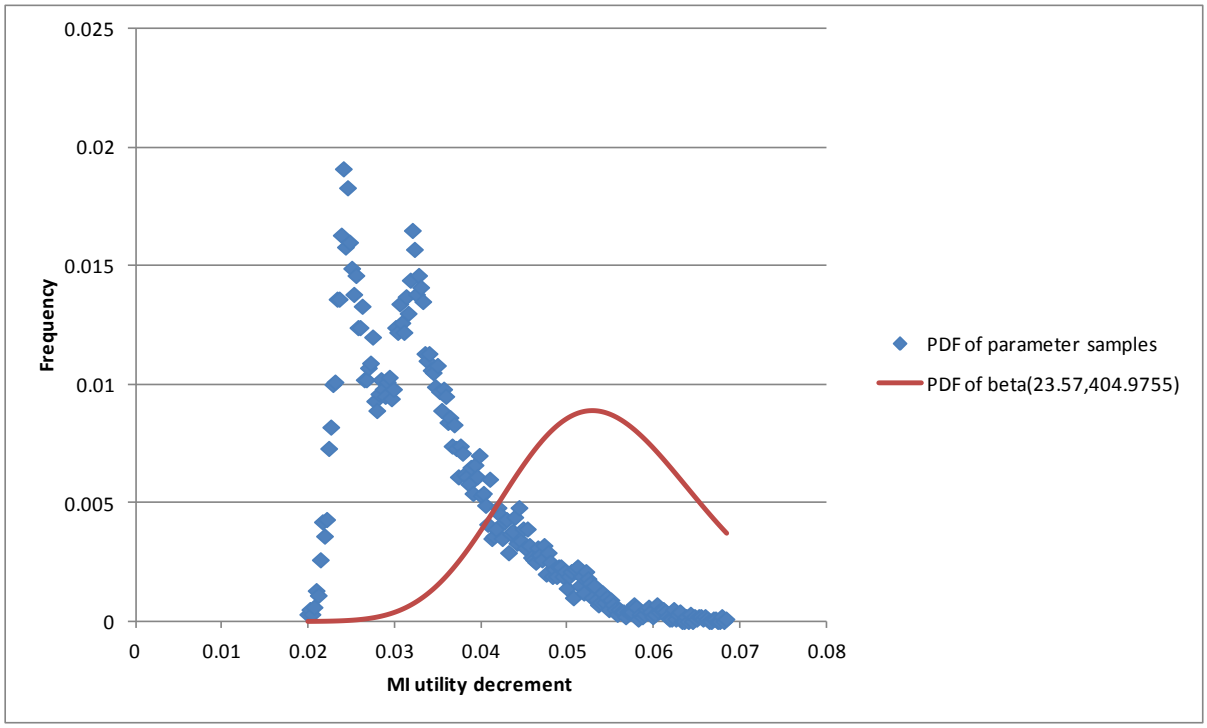
4.3. COMPARABILITY OF RESULTS GENERATED BY THE DIABETES2 AND DIAB2SAMPLING DLLS

The parameters to be varied within the PSA can be selected on the 'PSA map' sheet of the spreadsheet model. The DSU ran 100 cohorts of 30,000 using the probabilistic model but having specified no parameters to be sampled and compared these to the results produced from 100 runs of the model using mean parameter values. (In both cases the DLLs supplied in the original submission were used.) We would expect these results to be comparable but not identical, as although the same random number stream is being used in each case, the differences in C++ code between the two models allows for different numbers in that sequence to be used to sample the same event equation between the two versions of the model. The results, which can be seen in Appendix 3, are not comparable. The models are therefore not equivalent even when both use mean parameter values.

4.4. PARAMETER SAMPLING WITHIN THE PSA VERSION OF THE MODEL

Parameter samples (10,000) were extracted from the PSA version of the model and compared against the distributions specified in Table 72 of the MS for selected parameters. No problems were identified with the function used to generate parameters with a normal distribution (see Appendix 3). The function used to generate the parameters with a gamma distribution appeared to generate samples that fitted the distribution specified by the data in the columns labelled 'parameter 1' and 'parameter 2' in Table 72 (see Appendix 3). The samples also appeared to have an appropriate mean, but the standard deviation was approximately half that given in Table 72 for the parameter tested (standard deviation of 245 vs 449 for cost of fatal MI). The function used to generate samples for parameters using a beta distribution does not appear to provide the expected distribution (see Figure 8). The beta distribution (beta[23.57,404.9755]) given in Table 72 for the MI utility decrement is described as having a mean of 0.055 and a standard error of 0.0110. The samples extracted for this parameter had a mean of 0.033 and a standard deviation of 0.008. We also generated 10,000 estimates of the beta distribution specified in Table 72 for the MI utility decrement using the BETAINV function in Excel, and these were found to have a mean of 0.55 and a standard deviation of 0.011, as expected, suggesting that it is the method used to sample from the beta distribution within the model rather than the parameter values being passed into the model that is causing the error. The DSU also noted that the standard deviations for the age-dependent utilities at baseline are large compared to the difference between the mean values and the maximum value of 1 (e.g sd of 0.1754, mean of 0.877 for study 4) and this produced a highly skewed beta distribution which may not be realistic.

Figure 8: Comparison of parameter samples against specified distribution for the beta distributed parameter 'MI utility decrement'



4.5. RANDOM NUMBER GENERATION

The model uses the Mersenne Twister[2] algorithm to generate random numbers.As there was no documentation for this software or a version number that would enable discovery of any previously documented problems, two tests have been performed to examine whether it produces a high quality random number stream.

In the first test, data output from the algorithm was used to produce a black and white image which should look like "white noise" without obvious artifacts. Any obvious patterns would suggest a problem with the random number generator. A 1024x1024 pixel image containing 1,048,576 pixels was generated using the binary output from the model's random number generator. The output, which can be found in Appendix 2, showed no obvious patterns.

The second test used a specific RNG test suite called Dieharder.[3] The program takes a stream of random numbers from the random number generator and runs a series of tests on it. The results, which are provided in Appendix 2, are very good. Some areas were assessed as being 'weak' but these were either cryptography grade tests which are harder to pass or tests which have known faults.

5. CONCLUSIONS

The DSU found several differences between the description of the model provided by the manufacturer and the executable model submitted. The main differences are as follows;

- There are some discrepancies between the event equations and risk factor equations implemented within the model and those described by the manufacturer.
- Treatment related weight changes are applied immediately in the model rather than being achieved gradually over the first year of treatment.
- All cause mortality is not adjusted for stroke and MI fatalities
- The cost of renal monitoring is not being applied to all patients who start treatment with dapagliflozin.
- There are some discrepancies between the submission and the model with regards to the time periods over which some of the costs and utility decrements are applied.
- The process used to sample from beta and gamma distributed parameters within the PSA did not produce appropriately distributed samples

The DSU have also found some aspects of the model functioning that are not documented in the submission. These were as follows;

- The probability of an event occurring during a cycle is calculated as the difference between the output of the event equation for the current time and the output of the event equation at the previous cycle.
- Treatment discontinuations result in the patient switching treatment immediately and incurring no costs or health benefits from that treatment except for the discontinuation cost.
- The utility gain associated with BMI changes within the PSA model is based on the BMI profile generated using mean parameter values whilst the rest of the simulation uses a BMI profile that is sampled within the PSA.

The DSU were able to compile the DLLs from the source code provided. The diab2sampling DLL compiled by the DSU produced results identical to the diab2sampling DLL provided in the original submission. The DSU were unable to reproduce PSA results which matched

exactly those reported in the submission using the diab2sampling DLL, but the ICER generated by the DSU did not vary substantially from that reported in the submission and the differences may have arisen due to differences in the steps taken to set-up the PSA by the DSU as this required multiple changes to be made to the spreadsheet model provided. The diabetes2 DLL compiled by the DSU, which runs the model using mean parameter values, did not produce the results reported in the submission, and did not produce the results generated when using the diabetes2 DLL provided with the original submission. Furthermore, it did not appear to have reached a stable estimate of the incremental QALYs gained after 100 runs. The DSU also noted that the results generated by the diab2sampling DLL when all parameters are set to their mean value did not match those generated by the diabetes2 DLL which uses mean parameter values.

The DSU do not have confidence in the results produced by the model in its current form and would suggest that the following should be addressed as a priority. The version of the model that uses mean parameter values should produce results comparable to that produced by the PSA when all parameters within the PSA are set to their mean value. The process used to sample beta and gamma distributed parameters should be corrected to produce distributions consistent with those specified in the submission. The data used in the PSA for change in BMI from baseline should be based on the sampled BMI profiles. The DSU would also wish to see further clarification regarding the discrepancies identified between the event and risk factor equations implemented in the model and those reported in the MRCL.

6. REFERENCES

1. Clarke PM, Gray AM, Briggs A, Farmer AJ, Fenn P, Stevens RJ, *et al.*A model to estimate the lifetime health outcomes of patients with Type 2 diabetes: the United Kingdom Prospective Diabetes Study (UKPDS) Outcomes Model (UKPDS no. 68). *Diabetologia 2004*. 47(10):1747-1759.

2. Matsumoto M, Nishimura T. Mersenne Twister: A 623-Dimensionally Equidistributed Uniform Pseudo-Random Number Generator. *ACM Transactions on Modeling and Computer Simulation 1998*. 8(1):3-30.

3. Brown, Robert G., D. Eddelbuettel, and D. Bauer. Dieharder: A random number test suite. (http://www. phy. duke. edu/~ rgb/General/dieharder.php last accessed November 2012)

APPENDIX 1: FLOW CHARTS SHOWING MODEL LOGIC

See accompanying PDF file.

APPENDIX 2: RANDOM NUMBER GENERATOR TESTS

Figure A 1: Visual representation of the random number generator applied within the simulation (1024 x 1024 pixels)

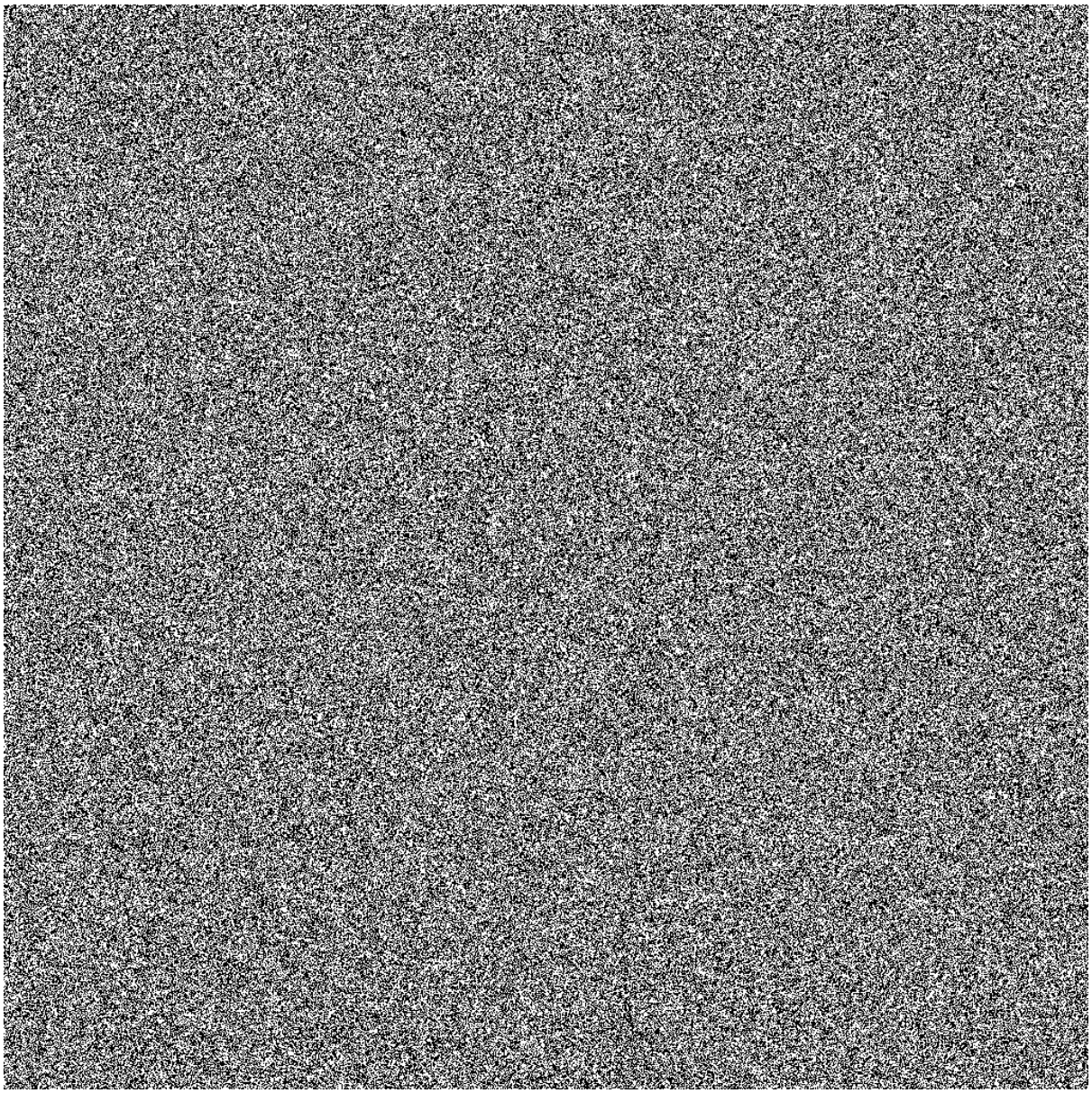


Table A 1: Diagnostic output from the Dieharder random number testing suite

```
#==========================================================================#
#            dieharder version 3.31.1 Copyright 2003 Robert G. Brown       #
#==========================================================================#
   rng_name    |rands/second|   Seed   |
stdin_input_raw|  1.33e+06  |1027554324|
#==========================================================================#
        test_name   |ntup| tsamples |psamples|  p-value |Assessment
#==========================================================================#
   diehard_birthdays|   0|       100|     100|0.34737130|  PASSED
      diehard_operm5|   0|   1000000|     100|0.05172491|  PASSED
  diehard_rank_32x32|   0|     40000|     100|0.75795923|  PASSED
    diehard_rank_6x8|   0|    100000|     100|0.97504041|  PASSED
   diehard_bitstream|   0|   2097152|     100|0.80562901|  PASSED
        diehard_opso|   0|   2097152|     100|0.99911775|   WEAK
        diehard_oqso|   0|   2097152|     100|0.76639111|  PASSED
         diehard_dna|   0|   2097152|     100|0.01933666|  PASSED
diehard_count_1s_str|   0|    256000|     100|0.44773774|  PASSED
diehard_count_1s_byt|   0|    256000|     100|0.82123469|  PASSED
 diehard_parking_lot|   0|     12000|     100|0.01724230|  PASSED
    diehard_2dsphere|   2|      8000|     100|0.65146473|  PASSED
    diehard_3dsphere|   3|      4000|     100|0.63559043|  PASSED
     diehard_squeeze|   0|    100000|     100|0.37239171|  PASSED
        diehard_sums|   0|       100|     100|0.92515086|  PASSED
        diehard_runs|   0|    100000|     100|0.90659512|  PASSED
        diehard_runs|   0|    100000|     100|0.72214482|  PASSED
       diehard_craps|   0|    200000|     100|0.86219490|  PASSED
       diehard_craps|   0|    200000|     100|0.84447234|  PASSED
 marsaglia_tsang_gcd|   0|  10000000|     100|0.06655969|  PASSED
 marsaglia_tsang_gcd|   0|  10000000|     100|0.69501561|  PASSED
         sts_monobit|   1|    100000|     100|0.51419073|  PASSED
            sts_runs|   2|    100000|     100|0.79369345|  PASSED
          sts_serial|   1|    100000|     100|0.96384593|  PASSED
          sts_serial|   2|    100000|     100|0.34884007|  PASSED
          sts_serial|   3|    100000|     100|0.85169495|  PASSED
          sts_serial|   3|    100000|     100|0.41150788|  PASSED
          sts_serial|   4|    100000|     100|0.12298543|  PASSED
          sts_serial|   4|    100000|     100|0.12997084|  PASSED
          sts_serial|   5|    100000|     100|0.30416596|  PASSED
          sts_serial|   5|    100000|     100|0.03214388|  PASSED
          sts_serial|   6|    100000|     100|0.13009854|  PASSED
          sts_serial|   6|    100000|     100|0.13800666|  PASSED
          sts_serial|   7|    100000|     100|0.91462829|  PASSED
          sts_serial|   7|    100000|     100|0.13997028|  PASSED
          sts_serial|   8|    100000|     100|0.86951231|  PASSED
```

```
           sts_serial|   8|    100000|     100|0.90981149|  PASSED
           sts_serial|   9|    100000|     100|0.85236232|  PASSED
           sts_serial|   9|    100000|     100|0.91526713|  PASSED
           sts_serial|  10|    100000|     100|0.73491536|  PASSED
           sts_serial|  10|    100000|     100|0.91348131|  PASSED
           sts_serial|  11|    100000|     100|0.21178049|  PASSED
           sts_serial|  11|    100000|     100|0.67237678|  PASSED
           sts_serial|  12|    100000|     100|0.62609526|  PASSED
           sts_serial|  12|    100000|     100|0.60872889|  PASSED
           sts_serial|  13|    100000|     100|0.85379815|  PASSED
           sts_serial|  13|    100000|     100|0.92274739|  PASSED
           sts_serial|  14|    100000|     100|0.44013440|  PASSED
           sts_serial|  14|    100000|     100|0.78778334|  PASSED
           sts_serial|  15|    100000|     100|0.73722762|  PASSED
           sts_serial|  15|    100000|     100|0.37805476|  PASSED
           sts_serial|  16|    100000|     100|0.55315527|  PASSED
           sts_serial|  16|    100000|     100|0.84842273|  PASSED
          rgb_bitdist|   1|    100000|     100|0.87159622|  PASSED
          rgb_bitdist|   2|    100000|     100|0.49709536|  PASSED
          rgb_bitdist|   3|    100000|     100|0.69539331|  PASSED
          rgb_bitdist|   4|    100000|     100|0.98100152|  PASSED
          rgb_bitdist|   5|    100000|     100|0.25568794|  PASSED
          rgb_bitdist|   6|    100000|     100|0.82626893|  PASSED
          rgb_bitdist|   7|    100000|     100|0.00189272|   WEAK
          rgb_bitdist|   8|    100000|     100|0.97630078|  PASSED
          rgb_bitdist|   9|    100000|     100|0.90387315|  PASSED
          rgb_bitdist|  10|    100000|     100|0.91729420|  PASSED
          rgb_bitdist|  11|    100000|     100|0.61294649|  PASSED
          rgb_bitdist|  12|    100000|     100|0.67611783|  PASSED
rgb_minimum_distance|   2|     10000|    1000|0.15458268|  PASSED
rgb_minimum_distance|   3|     10000|    1000|0.89766917|  PASSED
rgb_minimum_distance|   4|     10000|    1000|0.14652484|  PASSED
rgb_minimum_distance|   5|     10000|    1000|0.86253595|  PASSED
    rgb_permutations|   2|    100000|     100|0.16242912|  PASSED
    rgb_permutations|   3|    100000|     100|0.60361949|  PASSED
    rgb_permutations|   4|    100000|     100|0.89568308|  PASSED
    rgb_permutations|   5|    100000|     100|0.42813229|  PASSED
      rgb_lagged_sum|   0|   1000000|     100|0.26536819|  PASSED
      rgb_lagged_sum|   1|   1000000|     100|0.98316735|  PASSED
      rgb_lagged_sum|   2|   1000000|     100|0.82172364|  PASSED
      rgb_lagged_sum|   3|   1000000|     100|0.99836237|   WEAK
      rgb_lagged_sum|   4|   1000000|     100|0.23294807|  PASSED
      rgb_lagged_sum|   5|   1000000|     100|0.15332576|  PASSED
      rgb_lagged_sum|   6|   1000000|     100|0.60669675|  PASSED
      rgb_lagged_sum|   7|   1000000|     100|0.47204224|  PASSED
```

```
 rgb_lagged_sum|   8|   1000000|     100|0.85422061|  PASSED
 rgb_lagged_sum|   9|   1000000|     100|0.99382390|  PASSED
 rgb_lagged_sum|  10|   1000000|     100|0.77057057|  PASSED
 rgb_lagged_sum|  11|   1000000|     100|0.62809358|  PASSED
 rgb_lagged_sum|  12|   1000000|     100|0.71919071|  PASSED
 rgb_lagged_sum|  13|   1000000|     100|0.75097149|  PASSED
 rgb_lagged_sum|  14|   1000000|     100|0.58559878|  PASSED
 rgb_lagged_sum|  15|   1000000|     100|0.81454517|  PASSED
 rgb_lagged_sum|  16|   1000000|     100|0.99951799|   WEAK
 rgb_lagged_sum|  17|   1000000|     100|0.28902525|  PASSED
 rgb_lagged_sum|  18|   1000000|     100|0.98280608|  PASSED
 rgb_lagged_sum|  19|   1000000|     100|0.55647435|  PASSED
 rgb_lagged_sum|  20|   1000000|     100|0.46577303|  PASSED
 rgb_lagged_sum|  21|   1000000|     100|0.05264747|  PASSED
 rgb_lagged_sum|  22|   1000000|     100|0.43810486|  PASSED
 rgb_lagged_sum|  23|   1000000|     100|0.52275871|  PASSED
 rgb_lagged_sum|  24|   1000000|     100|0.99503035|   WEAK
 rgb_lagged_sum|  25|   1000000|     100|0.37380517|  PASSED
 rgb_lagged_sum|  26|   1000000|     100|0.60240597|  PASSED
 rgb_lagged_sum|  27|   1000000|     100|0.19278021|  PASSED
 rgb_lagged_sum|  28|   1000000|     100|0.10604288|  PASSED
 rgb_lagged_sum|  29|   1000000|     100|0.94581020|  PASSED
 rgb_lagged_sum|  30|   1000000|     100|0.17436078|  PASSED
 rgb_lagged_sum|  31|   1000000|     100|0.53764728|  PASSED
 rgb_lagged_sum|  32|   1000000|     100|0.68476148|  PASSED
rgb_kstest_test|   0|     10000|    1000|0.31193869|  PASSED
dab_bytedistrib|   0|  51200000|       1|0.15272894|  PASSED
        dab_dct| 256|     50000|       1|0.70743328|  PASSED
   dab_filltree|  32|  15000000|       1|0.85873453|  PASSED
   dab_filltree|  32|  15000000|       1|0.59932705|  PASSED
  dab_filltree2|   0|   5000000|       1|0.42691082|  PASSED
  dab_filltree2|   1|   5000000|       1|0.26863596|  PASSED
   dab_monobit2|  12|  65000000|       1|0.18731314|  PASSED
```

APPENDIX 3: PROBABILISTIC SENSITIVITY ANALYSIS VALIDATION TESTS

Figure A 2: Comparison of parameter samples against specified distribution for the normally distributed parameter 'HbA1c change' (DPP-4 in add-on to MET comparison)

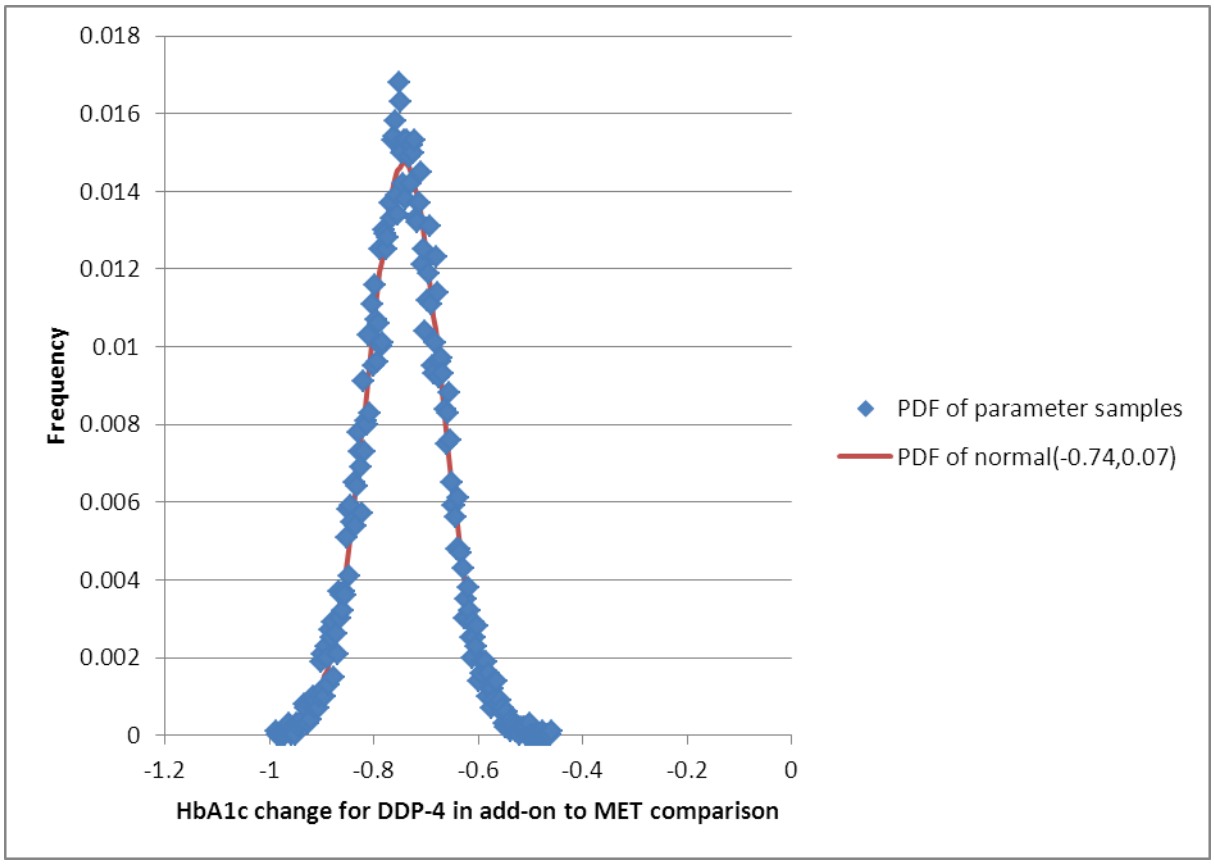


Figure A 3: Comparison of parameter samples against specified distribution for the gamma distributed parameter 'cost of fatal MI'

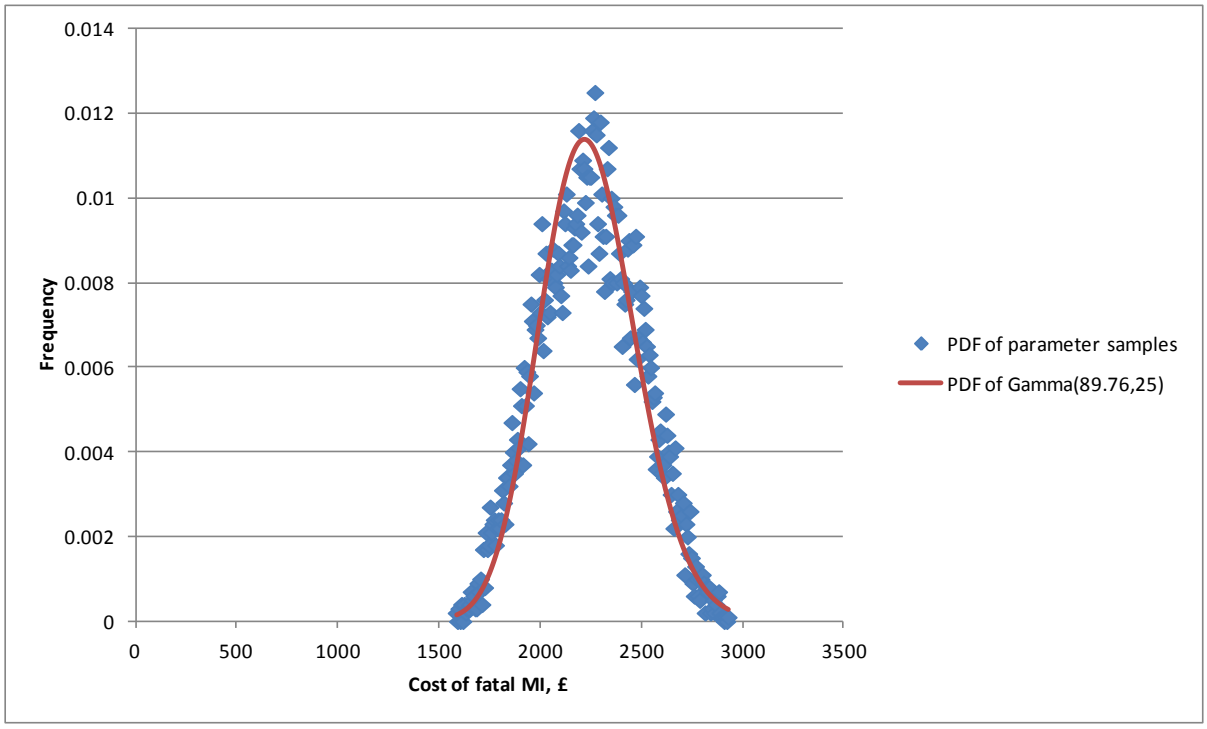


Figure A 4: Comparison of incremental QALYs when using the 'run model with mean values' option and when using the 'run probabilistic sensitivity analysis' option with no parameters sampled (dapagliflozin vs SU for add-on to metformin comparison)

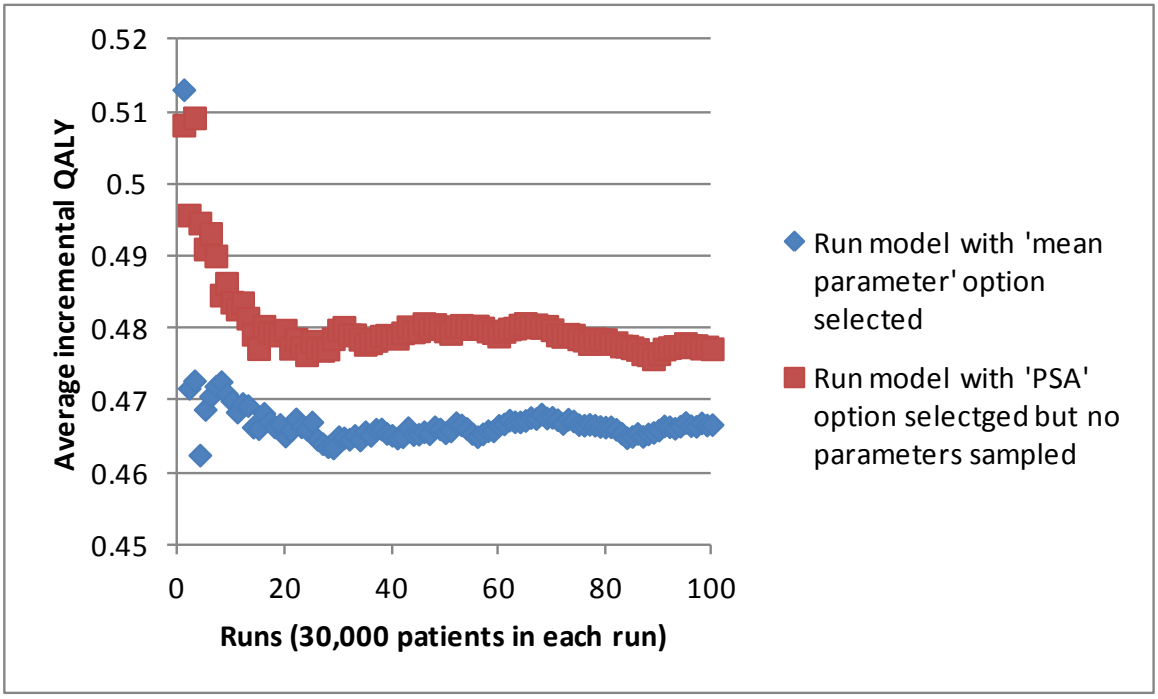


Figure A 5: Comparison of incremental costs when using the 'run model with mean values' option and when using the 'run probabilistic sensitivity analysis' option with no parameters sampled (dapagliflozin vs SU for add-on to metformin comparison)